Retrocomputer Magazine

Anno 3 - Numero 20 - Dicembre 2008

Come eravamo: Storia dei videogame

In prova: il kit NASCOM-1

BBS: Pubblicare una fanzine

TAMC: il quicksort

Apple Club: il FORTH

Il racconto: Jimmy 'er fenomeno

L'opinione: sciatteria all'italiana

ABAP: parte 8

Jurassic News

Rivista aperiodica di Retro-computing

Coordinatore editoriale
Salvatore Macomer [Sm]

Redazione
Sonicher [Sn]
redazione@jurassicnews.com

Hanno collaborato a questo numero:
Tullio Nicolussi [Tn]
Lorenzo 2 [L2]
Besdelsec [Bs]
Mister X [Mx]
Bob Lash

Impaginazione e grafica
Anna [An]

Diffusione
marketing@jurassicnews.com

La rivista viene diffusa in formato PDF via Internet agli utenti registrati sul sito www.jurassicnews.com.
la registrazione è gratuita e anonima; si gradisce comunque una registrazione nominativa.

Contatti
info@jurassicnews.com

Jurassic News promuove la libera circolazione delle idee

Dicembre 2008

Editoriale
Rosso Natale, 3

Retrocomputing
Ma il retro computing è difficile?, 4

Le prove di JN
NASCOM-1, 6

Come eravamo
Storia dei videogiochi (parte 1), 16

Laboratorio
Conservare le fotocopie, 22

Il racconto
Jimmy 'er fenomeno, 26

Edicola
HAX, Hax Are Experience, 28

Biblioteca
The BASIC Conversione Handbook, 30

Retro Riviste
Bit Games, 32

Retro Linguaggi
ABAP (parte 8), 34

DIR
Le facce del CP/M su Apple, 40

TAMC
Algoritmi di sort (parte 3), 46

L'opinione
Sciatteria all'italiana, 50

Apple Club
Tutti i linguaggi di Apple (parte 7), 52

BBS
Pubblicare una fanzine, 56

In Copertina

Le chiavi che aprono le porte della conoscenza sono fra loro intrecciate. Quando si apre una porta non si può mai sapere quante altre saranno di riflesso spalancate. Così è stato per le prime realizzazioni di un computer in kit: molti appassionati hanno trovato la strada per contribuire al progresso e alla conoscenza mondiale.

Editoriale

Rosso Natale,

scusate ma non abbiamo saputo resistere ad applicare un paramento di festa anche alla nostra rivista. Naturalmente un augurio sincero a tutti i retro computeristi d'Italia, che mantengano sempre intatta la loro passione.

Questo numero di Jurassic News si apre all'insegna di un ripensamento. Infatti per forza di cose ripieghiamo verso una periodicità meno puntuale, riservandoci semmai di rilasciare qualche numero "speciale" a completamento delle normali uscite.

Bando alle tristezze, ma non sono poi tali visto che se continua come finora la vita di JN sarà lunghissima, questo fascicolo ospita la prima parte di una delle tante storie dei videogame. Come sempre è una visione parziale e necessariamente lacunosa, ma l'argomento richiedeva senza ombra di dubbio che ce ne occupassimo, mentre chi vuole sa benissimo dove trovare tutte le notizie che desidera sull'argomento. Il nostro punto di vista è più "storico-culturale" che evolutistico, un approccio che ci si confà meglio che la mera esposizione della miriade di titoli che hanno caratterizzato fin dalle origini questo comparto dell'home computing.

Torna sulla rivista l'amico Maurizio con un'altro capitolo della sua "vita nella Megaditta", come al solito uno specchio sul reale che lascia un po' di amaro in bocca. Torna anche Alberta con una ficcante esternazione sullo stato della scuola italiana. E torna, o meglio prosegue la rassegna dei linguaggi per la piattaforma Apple: il FORTH, un linguaggio la cui sintassi sembra a prima vista studiata per scoraggiarne lo studio, ma che rivela insospettabili doti di sintesi e efficenza.

Con questo incompleto elenco di argomenti non mi rimane che augurare a tutti una piacevole lettura,

[Sm]

Jurassic News

è una fanzine dedicata al retro-computing nella più ampia accezione del termine. Gli articoli trattano in generale dell'informatica a partire dai primi anni '80 e si spingono fino ...all'altro ieri.

La pubblicazione ha carattere puramente amatoriale e didattico, tutte le informazioni sono tratte da materiale originale dell'epoca o raccolte (e attentamente vagliate) da Internet.

Retrocomputing

Dove ci si prepara al retro computing di domani.

Ma il retrocomputing è difficile?

La domanda che ci siamo posti è derivata da una discussione fra amici che verteva sulla consistenza della comunità di retrocomputeristi italiani.

Partendo dal nostro target, cioè da coloro che si sono iscritti al sito per scaricare la rivista e poi da quanti effettivamente lo fanno, unita all'analisi di quanto sono diffusi i siti Web italiani che parlano del nostro hobby, ci ha portati a stimare in non più di 200-300 le persone che in Italia fanno retro computing con una certa assiduità.

La stima è sicuramente sovradimensionata e vi possono cadere anche coloro che magari non sono proprio dei più assidui o non frequentano i gruppi di discussione, non curano siti web e così via. Considerando una popolazione di 60 milioni il numero è straordinariamente basso ma sale di qualche ordine di grandezza andando ad includere coloro che comunque il Commodore64 se lo tengono stretto o ne vanno alla ricerca su eBay o ancora giocano con l'Atari VCS e la collezione di cartridge che sono riusciti a mettere assieme.

Anche questi non sono tantissimi, sempre in rapporto alla popolazione, diciamo un 6.000 unità più o meno.

Questa stima l'abbiamo fatta computando la consistenza delle iniziative editoriali italiane e straniere che accentrano il loro interesse sul retro gaming.

La realtà dei numeri lascia sgomenti, soprattutto se consideriamo i milioni di sistemi venduti in venti anni in Italia. Lasciamo stare la diffusione attuale delle tecnologie informatiche, che poco ha a che vedere con il retro computing, ma gli stessi home di prima generazione (Apple II, Vic20, ZX80), quelli di seconda (C64, Spectrum, Atari ST) e per finire quelli di terza generazione (Amiga, Apple GS e Mac, etc...) sommano un venduto di almeno una decina di milioni di esemplari.

E' vero che magari sono sempre gli stessi che passano da una macchina alla successiva, ma alla fine degli anni 80 c'erano in Italia almeno 5 milioni di persone che usavano a casa una delle piattaforme home. Dove sono finiti costoro?

Qualcuno è morto, e va bene, fa parte della vita, ma alla fine possibile che solo un centinaio di essi abbiano conservato l'idea di continuare ad occuparsi di questi loro amici, magari vicini al nuovo PC, per riceverne il conforto della loro

vicinanza e della loro semplice fedeltà a confronto con l'instabilità della piattaforma PC che solo recentemente si è in qualche modo sdoganata dai famigerati schermi blu?

Senza voler bollare con il marchio dell'inferiorità coloro che considerano il retro gaming prioritario sulle altre attività similari è però indubbio che essi per la maggior parte si interessano poco di altro genere di software e frequentano quel minimo di hardware necessario ai loro scopi.

Questi sono sicuramente depositari di preziose esperienze, seppur verticalizzate su una o piattaforme, ma sono i "veri" retro computeristi, cioè coloro che hanno una esperienza più vasta, le persone che sono deputate alla conservazione dei sistemi e a tramandare la conoscenza in questo campo.

Non nascondiamolo: l'hobby del retro computing è difficile, incommensurabilmente lontano dalla collezione di tappi di bottiglie o regalini delle merendine. E' necessario un serio background tecnico per occuparsene e un certo impegno assiduo, tutte cose che sembrano appartenere più al passato che alla moderna mentalità del "tutto precotto" che coinvolge ogni aspetto della nostra vita sociale.

l'industria con la complicità del marketing ha trasformato le persone (lasciamoci fuori noi, almeno) in consumatori passivi invece che in attori attivi della tecnologia e del progresso.

Possedere un home nei primi anni '80 significava appartenere ad una elite e soprattutto partecipare in maniera attiva allo sviluppo della piattaforma: ci si scambiavano programmi ed esperienze, si scriveva qualche articolino sulle riviste, insomma si partecipava. Vi sentite adesso appartenenti alla comunità dei prodotti delle nuove tecnologie? Veramente condividete il possesso dell'ultimo modello di cellulare Nokia?

Anche dal rifiuto della "pappa pronta" nasce (o si conserva) la passione dell'occuparsi dei vecchi sistemi di calcolo. In questo nostro mondo le scoperte si sudano, magari si condividono, si assorbono da altri soggetti parimenti frequentanti la stessa tecnologia, ma sempre si accrescono con la cosidetta "santa pazienza".

Il numero esiguo di appassionati potrebbe sembrare una disdetta. Non è così: pochi ma buoni, verrebbe da dire, solo che una qualche migliore aggregazione sarebbe auspicabile se non necessaria, alla stregua di certe popolazioni di animali in pericolo di estinzione che possono trarre maggiori probabilità di sopravvivenza incrementando i fattori che ne favoriscono l'integrazione.

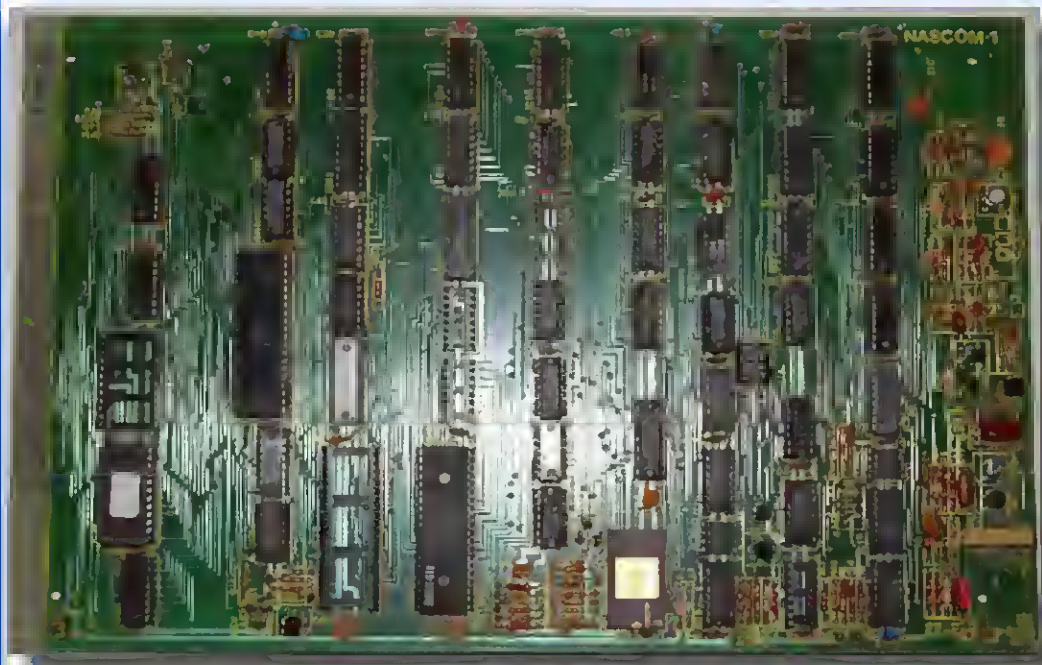
Per fortuna che c'è Internet!

[Tn]

Le prove di Jurassic News

Nasco Ltd. - NASCOM-1

Non solo in California furono forgiate le idee per un computer a basso costo, anzi da questo punto di vista la vecchia Inghilterra ha da sempre detto la sua. Ecco un progetto basato sullo Z80 che costa all'incirca 200 sterline (del 1978).

La piastra madre del NASCOM-1

Introduzione

Non solo gli States sono stati protagonisti della rivoluzione micro informatica. Infatti in misura minore ma non meno importante, anche altri paesi, come ad esempio l'Inghilterra e la Francia, hanno cercato una loro via.

Ad esempio questa piastra-Kit chiamata NASCOM-1 è un progetto britannico commercializzato dalla Lynx Electronics London, una sussidiaria della NASCO Ltd e concepito con l'intento di fornire ai primi appassionati un progetto didattico completo da usare come studio e base per la costruzione di un calcolatore personale completo.

Il progetto è del 1977, e segue quindi di pochissimo le prime piastre concepite in America. Sarà seguito dopo qualche tempo dalla versione 2, chiamata appunto Nascom-2, che evolve le funzionalità in direzione di un sistema di calcolo completo.

L'idea del progetto, presentata in due serie di articoli sulla rivista Wireless World nel 1977, venne ad un certo Shelton, CEO della Shelton Instruments Ldt, in collaborazione con il reparto progettazione della NASCO Ltd. Quest'ultima fu fondata da John Marshall dopo aver visitato la California ed avere visto il fermento creato dai vari Computer Club attorno ai progetti basati su microprocessore. Marshall, che lavorava per la NAS (North American

Semiconductors) concepì l'idea di realizzare un computer "general pourpose" in Kit e fondò pertanto la NASCO, poi chiamata NASCOM Microcomputers, per commercializzare l'idea. Il primo parto della società fu appunto il NASCOM-1, seguito purtroppo senza troppa fortuna dal NASCOM-2.

Il Nascom-1 è veramente minimale, senza cabinet o alimentatore e senza periferiche di storage se non la possibilità di collegarvi un registratore a cassette.

La dotazione software è parimenti striminzita, essendo nella versione base costituita da un monitor che occupa appena 1 Kb di ROM. Niente linguaggi di alto livello come il BASIC, ma ci si deve sporcare le mani con il linguaggio macchina: così si impara come funziona un microprocessore!

Il kit prevede una tastiera meccanica stile teletype e si dice che fu proprio questo componente a decretare il fallimento delle ambizioni di Marshall, dal momento che, esaurita la prima fornitura a basso prezzo, l'azienda non riuscì ad averne un'altra alla stesse condizioni, questo fece crescere di molto l'impegno finanziario e il prezzo del kit, inizialmente venduto a circa 200 sterline.

Si stima che furono vendute circa 12.000 kit nel giro di appena un anno, portando il NASCOM-1 all'attenzione di molte persone. Nacquero club, fanzine e riviste a supporto del progetto, che da questo punto di vista non ha nulla da invidiare ai computer progettati e costruiti nello stesso periodo negli States. Si dice che la prima rivista inglese di computer, Personal Computer World, sia stata voluta dall'editore proprio sull'onda del successo del NASCOM-1.

Primo approccio

Dal momento che il kit non prevede il cabinet, ci dobbiamo limitare ad uno sguardo alla nuda piastra madre che comprende tutta l'elettronica e al limite all'indispensabile tastiera che fa parte integrante del kit.

I componenti sono ordinatamente disposti su un circuito stampato di dimensioni generose (circa 30 x 20 cm) a doppia faccia. La distanza fra i chip e le varie piste di rame è adeguata ad un montaggio manuale anche se spreca dello spazio che renderebbe il sistema più contenuto nelle dimensioni.

Guardando la piastra con il con-

Un sistema NASCOM-2 presentato in occasione di un expo vintage.

La scheda madre rivolta con lo user-port verso l'alto. Come si vede si tratta di un esemplare privo della seconda ROM, mentre i socket da IC vuoti al centro della piastra sono i connettori per le due porte parallele e per la tastiera (quello isolato al centro).

nettore user-port verso l'alto si individuano subito sotto i due socket per le Eprom da 1 Kb (quello più a destra è riservato alle espansioni future). Subito sotto il vistoso chip della CPU Z80 e sul lato sinistro gli altri due socket da 40 pin per la UART e per il PIO.

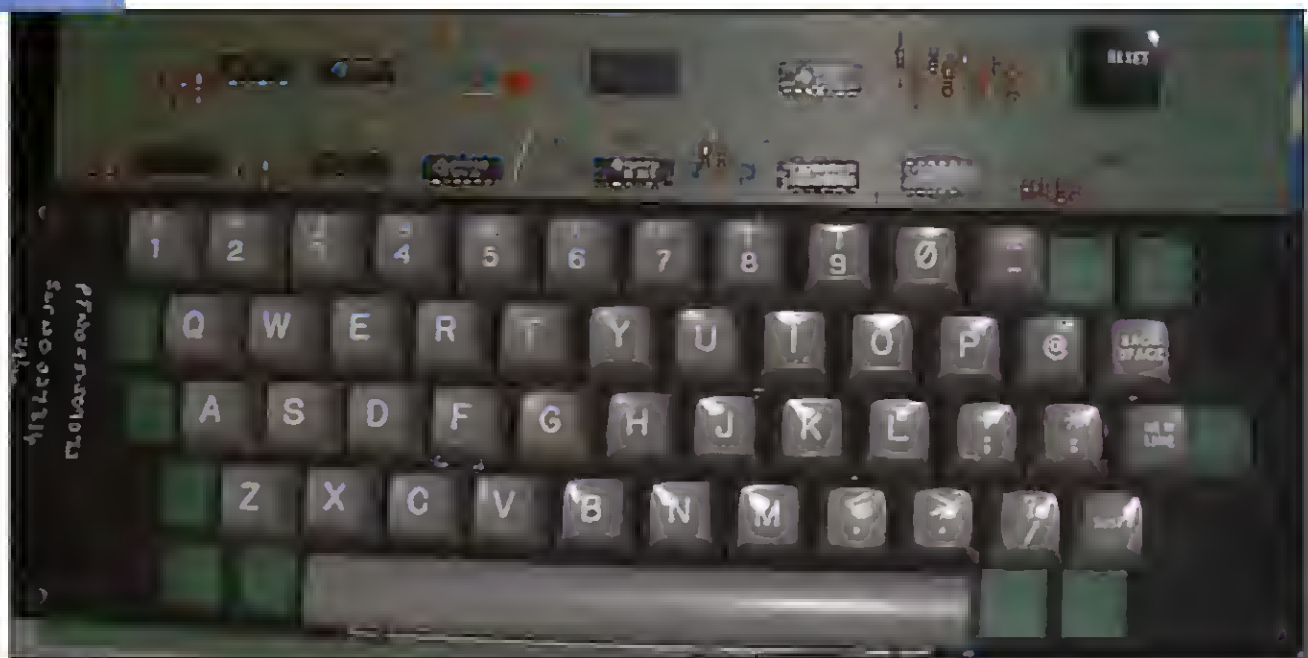
Il socket per la tastiera (un semplice zoccolo per IC da 16 pin), è montato in mezzo alla piastra, fra gli altri componenti, assieme ai due socket riservati alle due porte parallele gestite dal chip PIO. I chip di RAM, 16 AN2102 (1024x1 bit) sono curiosamente sparsi nella piastra, anche se si individuano i due gruppi corrispondenti ai due K disponibili e relativi semigruppi (semibyte alto e basso). Evidentemente ci hanno pensato parecchio e questa è la disposizione più conveniente dal punto di vista elettronico, anche se lascia sconcertati dal punto di vista organizzativo.

Peraltro anche i pin di alimentazione sono divisi ai due lati della piastra, una scelta davvero curiosa!

Nella parte bassa una serie di componenti analogici definisce il modulatore per l'uscita video in radio frequenza sul canale UHF. L'uscita prevede i 50 Hz per la frequenza di scansione, ma ne sono stati prodotti esemplari a 60 Hz adatti al mercato americano.

All'utente è lasciato il compito di sbizarrirsi riguardo al contenitore da adottare per il suo progetto, motivo per cui i NASCOM-1 cabinati sono tutti diversi fra loro.

Il problema del contenitore si complica quando si vuole adeguare la dotazione di RAM, con l'unica possibilità di affiancare una scheda esterna collegata via user port. La tastiera può essere ospitata in un contenitore esterno, facendo attenzione alla lunghezza dei cavi di collegamento che consistono in un cavo flat a 16 linee che porta anche l'alimentazione e il segnale di reset se si desidera predisporre un tasto apposito in prossimità della tastiera.

La tastiera del NASCOM-1. Si noti in alto a destra il tasto di Reset.

Hardware

Il kit è costruito attorno alla CPU Z80 funzionante a 1, 2 e 4 MHz, dispone di 2 Kb di RAM e 1 di ROM contenente un monitor di sistema "minimalista", presto upgradato a 2 KB con un "extended monitor".

Il fornitore della CPU è la Mostek con il suo source siglato MK3880, un fornitore che poteva permettersi un basso prezzo di vendita in stock.

In totale ci sono 55 chip e 10 transistor, più una serie adeguata di componenti passivi (resistenze e condensatori) con qualche diodo di disaccoppiamento e due zener fra le linee di alimentazione principali (+5 e +12Volt). Un cristallo oscillante a 18 MHz fornisce la stabilità necessaria al circuito di clock che sovraintende alla sincronizzazione dei vari componenti.

Le interfacce presenti on-board sono una seriale RS323, due parallele gestite da un chip PIO MK3881, quindi programmabili, una interfaccia TTY e una I/O per un registratore a cassette con velocità di 300 boud. La piastra comprende una uscita a radio frequenza per pilotare un monitor o un televisore adattato allo scopo grazie all'oscillatore UHF.

Il sistema di I/O seriale è coordinato da un chip UART cui fa capo il registratore e la porta RS232.

Un Kb di RAM è dedicata al video, per cui rimane uno striminzito Kb per i programmi utente, quantità che sarà ben presto messa in crisi all'approssimarsi di una maggiore capacità di sviluppo dei programmatori.

Ovviamente ci saranno modifiche ed espansioni in grado di portare la macchina verso i più decenti 16, 32 e anche 48 Kb di RAM, abbandonando la striminzita dotazione iniziale. A questo il progettista ha pensato in anticipo prevedendo una ponticellatura che esclude la RAM interna a favore di una eventuale espansione esterna da collegare alla porta utente.

Il chip ROM è una EPROM MK2708.

Sulla piastra è visibile un vistoso switch a tre posizioni che sceglie la velocità di clock del processore fra 1, 2 o 4 MHz.

La sezione video si avvale di un generatore di caratteri che usa buona parte del Kb di memoria video per mostrare a video 15 righe da 48 caratteri ciascuna. Le righe indicizzate in RAM sono in realtà 16, una viene usata come buffer e "scrollata" a cura del programma monitor.

CPU e sezione video si contendono l'accesso alla RAM e pertanto possono incappare in conflitti che sono risolti a monte dando priorità alla CPU. Questo significa che il video sarà soggetto a "black-out" in occasione delle operazioni che la CPU effettua sulla memoria video, che è poi una mappatura della RAM di sistema.

Una UART IM-6402CPL è dedicata al comparto periferiche seriali, compresa la tastiera e il registratore a cassette. Il progetto prevede il supporto per un lettore/registratore a nastro perforato, retaggio dei tempi passati.

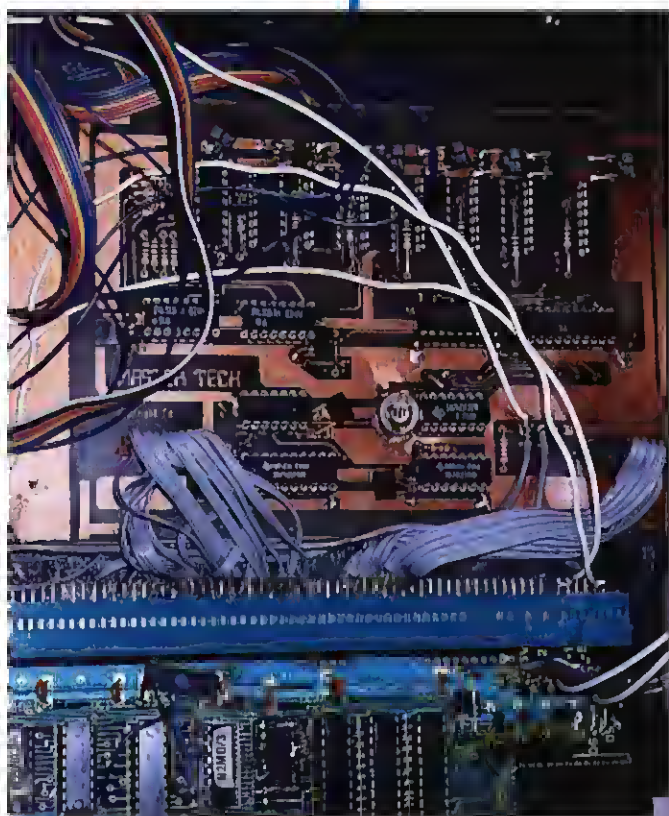
L'oscillatore video, capace di pilotare un televisore in UHF, ovviamente in bianco/nero, è un circuito molto semplice: infatti usa un solo transistor, un diodo, una impedenza e una manciata di componenti passivi.

La tastiera è una classica QWERTY, collegata attraverso la sezione I/O seriale che fa capo alla UART e prevede anche un pulsante di reset collegato direttamente alla corrispondente linea della CPU.

Non manca il connettore "User Port", realizzato con terminazioni del circuito stampato e che occupa praticamente un intero lato "corto" della piastra con le sue 43 terminazioni disponibili a future espansioni.

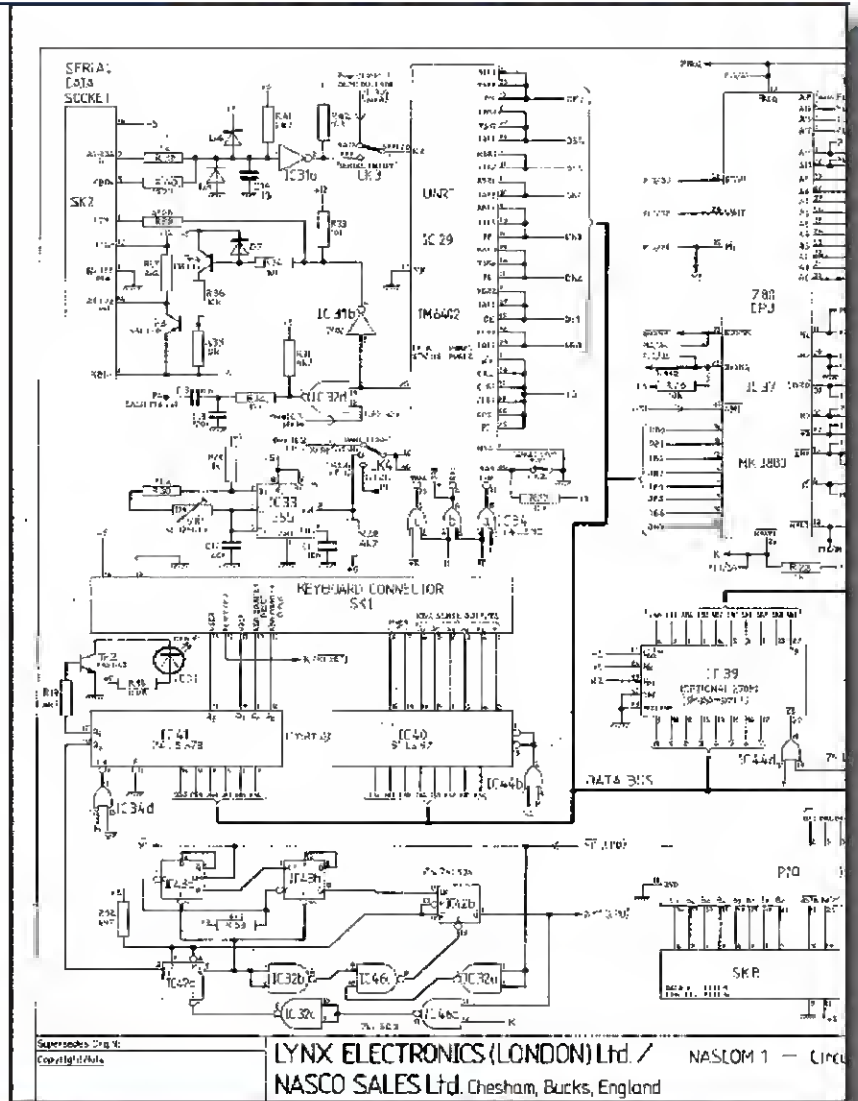
Il sistema viene fornito in Kit o montato ma in ogni caso è cura dell'acquirente provvedersi di alimentatore (+5, -5, +12, -12), di registratore a cassette, monitor o TV e se lo desidera di un cabinet adeguato.

La mappatura della memoria prevede la ROM posizionata negli indirizzi da pagina zero in poi. La prima EPROM da 0000h a 03FFh e un successivo spazio da 1 Kb che fa capo ad uno zoccolo libero sulla piastra per future espansioni. Segue la RAM video: da 0800h a 0BFFh , mentre dall'indirizzo 0C00h in poi viene posizionata la RAM utente.

In questa foto la user-port è stata collegata in maniera "volante" ad una espansione di RAM.

La scelta che prevede la ROM negli indirizzi bassi di memoria è utile perché lascia libera l'espansione di RAM futura ma in qualche modo incatena la gestione delle routine di base del processore che fanno riferimento ad indirizzi in pagina zero: tipicamente gli indirizzi di salto per gli interrupt.

Benché i principali circuiti integrati siano provvisti di zoccolatura nel kit, nondimeno è necessario acquistarne a parte se si vuole gli zoccoli anche per i chip "meno delicati". L'alimentatore non è fornito di serie ma naturalmente viene venduto opzionalmente in un kit separato. Infine una serie di jumpers sono predisposti per la configurazione di alcuni parametri come il clock e l'utilizzo della RAM esterna se presente.

Non sono presenti connettori di output, ma semplici terminazioni sulla piastra o ai pin di alcuni zoccoli (come per la tastiera). L'utente deve provvedere da se a comprarsi eventuali prese per il video e il registratore (che mi sembra opportuno non siano collegati in maniera inamovibile alla piastra. Per il registratore viene consigliata la costruzione di un semplice circuito di dumping del segnale per rendere il sistema meno sensibile alla qualità del registratore domestico che si intende usare. La periferica registratore a cassette non è comunque critica anche grazie alla bassa velocità di registrazione: appena 300 boud.

Trattandosi di un Kit, particolare cura è stata messa nella completezza della documentazione con schemi elettrici e addirittura con le maschere del circuito.

Uso

Siamo difronte, come dicevamo, ad un sistema "primitivo" e adatto soprattutto come oggetto di studio e sperimentazione dell'elettronica digitale a microprocessori.

Al momento dell'accensione è il monitor di sistema che prende il controllo e mette a disposizione alcuni semplici comandi per interagire con la memoria e con le periferiche standard. E' comunque un sistema più evoluto rispetto ai kit da laboratorio, programmabili il più delle volte con un tastierino esadecimale. Tastiera e video sono gestiti correttamente dal semplice software di sistema e il caricamen-

British Home

MICROCOMPUTER

NASCOM 1

SOFTWARE NO

NASCO
SALES LTD

LYNX Electronics (LONDON) LTD.

92 BROAD STREET
CHESHAM, BUCKS.

Tel: Chesham (02405) 75151
Telex. 837571

la copertina di uno dei manuali in dotazione

Una realizzazione casalinga del Nascom-1

to da cassetta di un macro assembler, ad esempio, rende completo e utile il computer.

Assieme al kit viene recapitata una semplice documentazione che comprende schemi elettrici, note di costruzione e i primi rudimenti della programmazione in linguaggio macchina.

Sulla prima pagina del manuale si evidenzia l'origine britannica del kit con un bel "NASCOM-1 British Home Microcomputer" a ribadire che si tratta di un prodotto del vecchio continente e non proveniente da oltre oceano. Evidentemente l'orgoglio nazionale era ancora presente e forse poteva determinare una quantità di venduto superiore rispetto ad un simile prodotto di provenienza estera.

Un'altra curiosità che si legge nel manuale è l'invito all'utilizzatore ad associarsi ai nascenti Computer Club, per ricavare il meglio dalla sua macchina. Una accortezza che oggi fa ridere ma che evidentemente è fondamentale per un mercato che sta nascendo dal nulla o quasi.

Il manuale dedicato alla costruzione del kit si prolunga nei soliti consigli su come sia meglio saldare i componenti o gli zoccoli, l'attenzione da spendere per verificare che non ci siano cortocircuiti fra le piastre, i primi controlli a fronte del mancato funzionamento, etc...

Il manuale giudica necessarie 20 ore per completare la costruzione, consigliando anche di non spendere più di due o tre ore di fila sul kit, ma di diluire la realizzazione su più giornate. Un consiglio paternalistico ma sicuramente valido ad evitare eccessivo stress nella realizzazione di un circuito certo non fra i più facili per un hobbista e che termina con un buon augurante "Good luck!".

Il generatore di caratteri è in grado di rendere i 128 simboli codificati nella ROM (che può essere localizzata se lo si desidera, esiste anche il giapponese! Certo che erano di larghe vedute questi della NASCO!).

I caratteri (maiuscoli, minuscoli e simboli vari) sono codificati in una matrice di 9 righe per 7 colonne e garantiscono una buona visibilità a video.

Software

Quello che viene chiamato pomposamente “Operating System” dalla NASCO, altro non è che un monitor di sistema che trova posto nello striminzito spazio della ROM.

Il NASBUG, così si appella il programma, prende il controllo del sistema al reset e visualizza il prompt sull’ultima riga del video, il carattere “>”, pronto ad accettare i comandi utente. Questi sono costituiti da una sola lettera seguita eventualmente dai parametri. Il NASBUG permette di editare il comando prima dell’invio utilizzando il tasto di back-space. La conferma avviene al solito con il New-Line.

I comandi sono quelli tipici che servono per assemblare il codice in esadecimale, senza assemblatore o altro:

Mxxxx : visualizza/modifica la memoria a partire dall’indirizzo xxxx (esadecimale);

Txxxx yyyy : dump della memoria a video dall’indirizzo xxxx all’indirizzo yyyy;

Cxxxx yyyy zzzz : copia un blocco di memoria di lunghezza zzzz dall’indirizzo xxxx all’indirizzo yyyy;

Exxxx : esegue il codice a partire dall’indirizzo xxxx;

Bxxxx : setta un breakpoint all’indirizzo indicato;

Sxxxx : esegue un single-step di programma (se l’indirizzo è omesso viene eseguito lo step successivo);

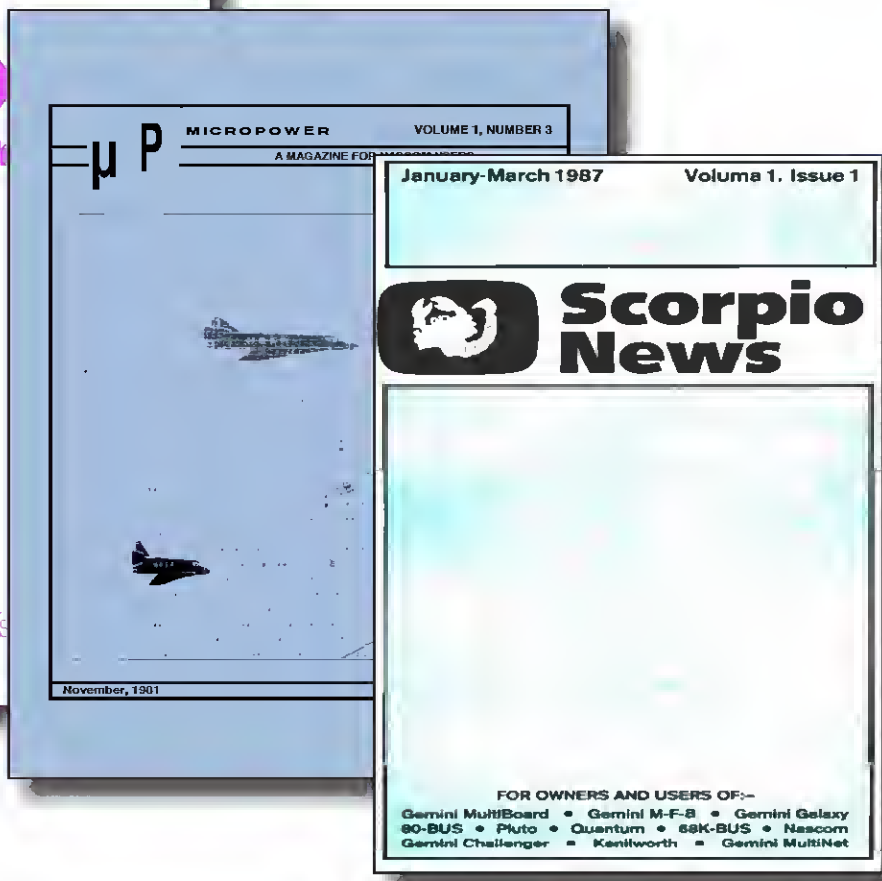
A mano a mano che vengono inseriti comandi il NASBUG provvede allo scroll del video: la riga “operativa” è sempre l’ultima, per cui il contenuto del video stesso è puramente documentativo e non può essere indirizzato dalle operazioni di programmazione. Ovviamete l’esecuzione di un programma complesso che utilizza la memoria video come output, ne userà a piacere l’area disponibile, tornando poi sull’ultima riga con il prompt al momento della ri-esecuzione del NASBUG.

L’esecuzione single-step comporta il display a video del contenuto dei registri della CPU.

Il NASBUG deve per forza di cose utilizzare parte della RAM utente per appoggiare i propri dati e infatti dall’indirizzo 0C00h a 0C4F, si trovano 64 byte riservati al NASBUG stesso, quindi i programmi utente possono partire dall’indirizzo 0C50h.

A proposito dell’area NASBUG, una parte è riservata come specchio dei registri della CPU. Quindi se si vuole eseguire un programma con una certa configurazione dei registri è necessario, prima del comando E, predisporre i relativi valori nelle locazioni riservate. Una modalità abbastanza scomoda, ma tutto sommato di facile memorizzazione dopo un breve periodo

Gli User Group si sostituiscono alle aziende e agli editori specializzati per supportare un prodotto di successo ma non al punto di invogliare l'editoria ufficiale.

La copertina del primo numero di PCW, aprile 1978. La rivista venne concepita da Angelo Zgorelec, un croato che viveva in GB.

di utilizzo.

Disporre dell'area "buffer" del NASBUG è comodo per controllare tutti gli aspetti del programma monitor, ad esempio se si vuole controllare l'effetto di un dato proveniente dalla tastiera senza digitare un tasto, si inserirà il valore corrispondente direttamente nell'area RAM predisposta a questa funzione.

Il manuale referenzia le chiamate alle routine più utili per la costruzione dei propri programmi, come l'output a video e la scansione della tastiera.

La NASCO annuncia la disponibilità di un "Tiny Basic" da 2 Kb e della futura disponibilità di altri linguaggi di programmazione, come il FORTRAN ad esempio, previa costruzione dell'espansione di RAM, che all'epoca era costosa e probabilmente abbastanza lontana dalle possibilità economiche di molti.

[Tn]

June/July/August 1980

INMC NEWS 80 Issue: 1

c/o Oakfield Corner, Sycamore Road, Amersham, Bucks HP6 6SU

BUMPER "We're still around" ISSUE

CONTENTS

Page 1 This is It.
Page 2 Chairman's Bit.
Page 3 Long Live NASCOM.
Page 4 Letters to the Editor.
Page 10 NASCOM Rescue Bid.
Page 11 Print HL in Decimal.
Page 12 Checksum Routines.
Page 14 ZEAP Output to Printer.
Page 15 Conversion of T4 to Nas-Sys.
Page 16 Make Your Own Characters.
Page 18 Hex Codes/Graphics/Reserved Words.
Page 19 Teach Yourself Z80.
Page 22 S100-Readers' Replies.
Page 24 Using the PIO.
Page 26 Hardware Reviews.
Nascom I/O Board.
Page 28 Port Probe & Key Pads.
3M Cassettes.
Page 29 CUTS Cassette Interface.
Page 30 BASIC at 4MHz.
Page 31 Notes on PASCAL.
Page 32 Book Reviews.
Z80 Books.
Page 33 BASIC Books.
Classified Ads.
Page 34 Doctor Dark's Diary.
Page 36 WHAT ?
Page 37 Index to INMC Newsletters 1 - 7.
Page 39 Computers ?
Page 40 Assemblers and Nas-Sys.
Page 41 HALT and the Nascom.
Page 42 Generated BASIC Programs.
Page 44 Cassette Hints.
Cassette Reliability.
Cut That Noise.
Page 45 2400 Baud on Nascom 2.
Page 46 ZEAP 1.1 Files to 2.0 Files.
8 Amp PSU - A Warning.
Page 47 Nas-Sys 3.
Page 48 Laurence Fights Back.
Page 49 Our very own "IMPERSONAL" column.
Pages 43/50/51 Advertisements.

PLEASE NOTE. Our address is now:
INMC80, c/o Oakfield Corner,
Sycamore Road, Amersham, Bucks. HP6 6SU.

Non-members 90p

PLEASE ALSO NOTE. This address is used by INMC80 purely as a postbox. We cannot arrange to send you leaflets or answer all your sales and technical queries. Contact NASCOM or your NASCOM Distributor for these.

inmc news issue 1

Editorial

This is the first real INMC Newsletter, at last. I'm sorry it took so long to get the ball rolling. This issue is mostly in the form of "messages from our sponsor" but we hope this will change and that you, the members, will contribute the majority of the contents in future. We would like to hear from you if you would like to

- contribute a brief write up of an interesting application
- place a small ad (free to INMC members)
- start a local computer club
- heap praise (or blame) on us
- submit interesting circuits (e.g. for interfacing things)
- submit useful programs
- do anything else

Most programs will still be distributed in the usual way, but the smaller ones of general interest will be published in the Newsletter.

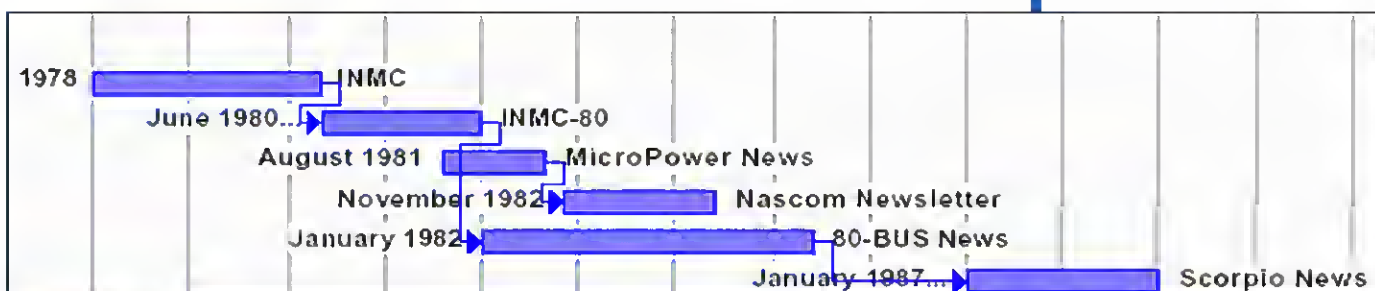
Please also let me know if you have any comments about the Newsletter format. Is it what you want? If so, tell us - it makes us feel wanted. If not, tell us so we can change it - the INMC is for you, not us.

I would like your advice on one particular point. It has been suggested that we publish a list of INMC members to help you get in touch with each other locally. This seems a good idea, but the list might then be used by anyone who gets hold of it for mail shots. This could increase the number of unwanted advertising material you get through the post. What do you think?

Having started, we hope to continue to to issue the Newsletter at approximately six weekly intervals. I hope you like it.

A.R HINDLE

Il Gantt nella figura riassume la sequenza di rilascio delle fanzine dedicate al Nascom-1.

Come eravamo...

Storia dei videogiochi (parte 1)

La storia dei sistemi e degli uomini che hanno creato un mondo nuovo.

Dalla disarmante semplicità di PONG alla sconvolgente esperienza di DOOM3, la storia dei videogame è lunga più di vent'anni. In questo articolo ne percorreremo le tappe principali, senza avere la pretesa di essere esaustivi, cosa che richiederebbe ben altro impegno.

La storia dei videogame non coincide con quella dei personal computer, anche se le due si sono incrociate molte volte e hanno percorso lunghi tratti in parallelo. Pensiamo ad esempio al fenomeno degli arcade, le piastre e cabinet costruiti appositamente per quel settore, oppure alle console specializzate che niente hanno a che fare con l'idea del computer personale per uso generale.

Se esaminiamo il fenomeno dell'evoluzione dei sistemi per giocare sul video (non per nulla si chiamano "videogiochi"), vediamo che la storia degli stessi può essere ripercorsa considerando le innovazioni tecnologiche che sono legate proprio all'aspetto di presentazione e, più recentemente, di coinvolgimento reale (vedi la WII machine della Nintendo).

I primissimi giochi che potevano essere effettuati con un supporto/coinvolgimento di una macchina di calcolo, risalgono all'epoca dei mainframe installati a partire dal dopo guerra nelle grandi università o enti governativi come il Dipartimento della Difesa (DoD) negli Stati Uniti.

Moltissimi programmi si vantano di essere stati i primi e questo deriva dalla percezione che le persone hanno rispetto all'evento con il quale sono entrati in contatto con questo mondo o ne hanno sentito parlare. Se si fa una indagine statistica fra le persone chiedendo loro di indicare quale è stato il primo videogame, la maggioranza indicherà PAC-MAN o forse Space

La console Intellivision, forse la prima macchina da gioco casalinga che si ricordi.

Invaders.

Questo non è vero dal punto di vista storico, infatti prima che apparisse la pallina gialla mangia-puntini, introdotta dalla Midway nel 1980, o anche il più semplice spara-agli-alieni, ci sono stati una decina d'anni di console specializzate e almeno 30 anni di giochi sui mainframe.

Prima di tutto: qual è stato il primo videogame in assoluto? Non vi sorprenderete nell'apprendere che il gioco che per primo è stato rivisto in versione elettronica è il famoso tic-tac-toe, conosciuto anche come ics-e-zero e in Italia "Filetto" o "Tris".

Dalla nascita dei sistemi di calcolo, anche meccanici, ci sono state versioni più o meno automatiche di giochi matematici o logici, ma mancava la parte "video", essenziale per farli annoverare nella categoria.

Un certo Douglas, ricercatore alla Cambridge University, ne realizzò la prima versione elettronica nel 1952 come dimostrazione della possibilità di interazione uomo-macchina. Dato che la sua versione prevedeva un display a video su terminale CRT, questo deve essere considerato il primo videogioco in assoluto. Il gioco girava su un calcolatore EDSAC ma, come spesso accade, il riconoscimento di inventore del videogioco non è stato dato a lui ma a William Higinbotham che nel 1958 creò "Tennis for two", una primitiva versione di Pong che usava un oscilloscopio come terminale di output. La realizzazione avvenne presso il Brookhaven national Laboratory di New York. Il display mostrava il campo di gioco e una pallina soggetta anche alla forza di gravità, quindi con un moto simulato abbastanza precisamente e ovviamente due "racchette" manovrate da altrettanti giocatori. La realizzazione pratica del controllo era un potenziometro e un pulsante per dare "il colpo" alla pallina e inviarla oltre la rete.

Per l'Intellivision sono stati commercializzati un centinaio di giochi, la grafica non era un granché, ma ci si accontentava...

Il gioco godette di una certa popolarità fra i visitatori del National Laboratory dove rimase in esposizione per circa un anno.

Questi primi esperimenti di interazione uomo-macchina nel settore ludico diedero la stura ad una miriade di progetti realizzati dagli studenti delle varie università che avevano a disposizione un mainframe.

La storia vuole che il primo nucleo del sistema operativo Unix sia stato sviluppato da Ken Thompson per disporre di un ambiente adatto a giocare una versione di Space Traver sul PDP-11 installato al Bell Laboratory.

Possiamo sorprenderci che una innovazione tanto pregnante per la moderna industria informatica sia stata sviluppata "per gioco", ma del resto moltissime delle innova-

Da una immagine pubblicitaria dell'epoca l'Atari 2600, completo di unità di espansione.

zioni che consideriamo normale dotazione di un PC, sono state realizzate per scopi ludici. Ad esempio le schede sonore sono evidentemente derivate dalla necessità di riprodurre meglio il sonoro dei giochi rispetto a quanto fosse possibile con il semplice beeper dei primi personal; a nessuno sarebbe venuto in mente di usare un PC come impianto stereo! La stessa tecnologia delle schede grafiche a colori prima e 3D poi, derivano dalla spinta del mercato entertainment.

L'esplosione dei videogiochi come fenomeno di massa avviene a partire dal 1972 quando la Magnavox introduce la console chiamata Odyssey. La console si connette direttamente al TV di casa ed è realizzata con circuiti analogici, assemblata con oltre 300 pezzi ed utilizza una grafica che possiamo definire "posticcia" nel vero senso della parola, visto che ogni gioco prevede l'applicazione di un foglio trasparente sullo schermo TV per la riproduzione dello sfondo grafico del gioco. Per questo nuovo oggetto di intrattenimento casalingo sono disponibili 12 titoli, con relativi "sfondi", ma non Pong! Perché? Vi chiederete. La risposta è che la guerra del copyright era già cominciata.

Pong viene venduto come console stand-alone da attaccare al TV che diventa il vero best-seller del settore. L'inventore è Nolan Bushnell, che per la commercializzazione della sua invenzione fonda la società Atari. Pong è stato venduto in 19.000 esemplari ma questo numero aumenta a dismisura se andiamo a contare tutte le imitazioni che ne sono seguite.

L'idea di disporre di un corpo macchina di base con la possibilità di inserire varie cartucce per cambiare gioco, viene sfruttata dall'Atari che nel 1977 rilascia il VCS (Video Computer System), noto anche come Atari 2600. Con questo prodotto l'Atari conquista decisamente la leadership del mercato, surclassando il vero inventore della console "a cartucce", la Farchild con il suo Channel F (1976).

L'importanza della disponibilità

delle cartridge che permettono lo scambio dei giochi è essenziale. Ci sono due aspetti fondamentali: il primo è l'idea della console multi-game, cioè la console funge da base per molti titoli che i programmatori possono rilasciare migliorando costantemente l'esperienza di gioco. Questo significa che per l'acquirente l'oggetto viene ad assumere un valore superiore proprio perché "flessibile". Il secondo aspetto è la possibilità per gli utenti di scambiarsi i titoli e di realizzare una vasta libreria di game, che posso anche scambiare con amici e parenti. La vendita della base è solo uno degli aspetti del marketing del prodotto; il vero business si realizza nella vendita dei programmi e disporre di un hardware diffuso e con molti titoli a disposizione rafforza la posizione di leadership di chi possiede questa chiave.

La disponibilità di una macchina "aperta" stimola la progettazione di accessori di terze parti e delle case di software che si dedicano alla realizzazione delle cartridge. Una delle prime è la Activision, che diventerà un vero colosso del settore.

Questa fase della storia dei videogame viene classificata come "8-bit era", anche se certe realizzazioni, come la console Intellivision della Mattel, avevano un processore a 10 bit.

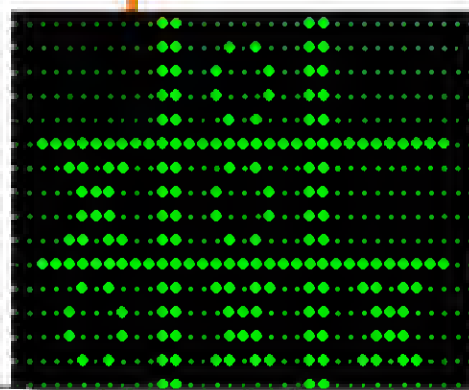
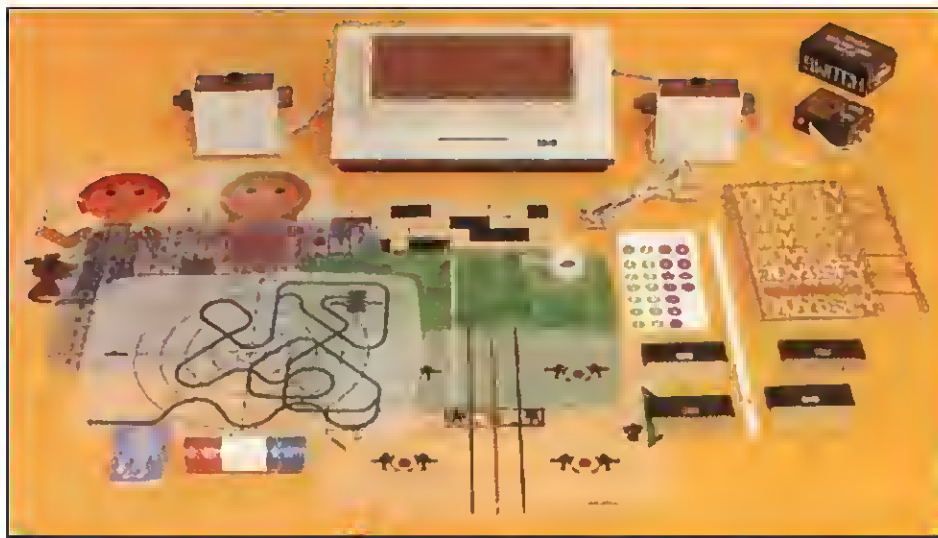
Per i programmatori la disponibilità di 4Kb nelle cassette Intellivision e addirittura solo 2Kb in quelle Atari 2600, erano una sfida davvero notevole e che ben presto avrebbe rivelato tutti i suoi limiti.

Pong è sicuramente il primo videogioco di massa e per il quale sono state costruiti cabinet dedicati.

All'inizio del 1980 un altro fenomeno segna la storia dei videogiochi: l'apparizione degli "Arcade", cioè di quelli che vengono anche definiti "giochi da bar". Da sempre il settore dell'intrattenimento pubblico è stato fiorente perfino in situazioni di depressione economica. Un mio conoscente, operatore del settore, mi raccontava come durante la seconda guerra mondiale con la borsa nera e tutto il resto, dove la gente non riusciva a mettere assieme il pranzo con la cena, anche allora la gente giocava. Evidentemente è più forte di qualsiasi altro bisogno!

Sta di fatto che cominciano ad apparire nei bar, accanto al classicissimo bigliardino, anche strani cabinet con video (inizialmente in bianco/nero), massicci joystick e pulsantoni di plastica. Ovviamente non manca la gettoniera, così che con 100 lire ti fai una partita a Space Invaders, un ossessionante giochino spara-alieni capace di tenere incollati al video per ore parecchi adolescenti.

Duro colpo per il flipper, icona e amore di intere generazioni! Ma non c'è

Una vecchissima versione del "Tris" o "Tic-Tac-Toe". Come output si usava il display di un oscilloscopio.

Il "sistema" Odissey commercializzato dalla Magnavox a partire dal 1972.

niente da fare, fra l'altro l'arcade occupa meno spazio nel locale e quindi è preferito anche dai gestori. Il flipper non si risolleverà più da questo duro colpo; sì qualcosa rimane ma dal '90 è decisamente un settore marginale.

Dal 1980 al 1985 possiamo individuare l'esplosione dei cabinet nei bar e la nascita delle sale giochi con decine di questi aggeggi che emettono suoni "spaziali" come sibili ed esplosioni per non parlare del rumore di rombanti motori su piste immaginarie. I cabinet sono costruiti ad-hoc, spesso fantasiosamente customizzati e sagomati a guida di navicelle spaziali o altro. All'interno si trovano le cosidette "piastre" che ospitano la componentistica, spesso a due o più piani, che si fanno sempre più complesse e sofisticate.

Si fa risalire a questa epoca la nascita di nuovi generi in una continua corsa delle aziende al continuo tentativo di superarsi.

La numerosità dei giochi è frutto anche di un fenomeno imprevedibile: la gente si stufa di giocare sempre lo stesso titolo e di conseguenza nei locali è necessario sostituirli ogni tre mesi almeno. Questo fenomeno è stranissimo se consideriamo che altri giochi "da bar" sono per loro stessa natura eterni: il calcetto ad esempio e anche lo stesso bigliardino (flipper) viene cambiato finora con frequenza bassissima (tre anni sono la norma).

L'apparire sul mercato di qualcosa di diverso provoca degli scossoni. Ad esempio il gioco "Defender" uscito nel 1980 segna una svolta sotto il profilo del coinvolgimento di gioco e della necessità per il giocatore di tenere sotto controllo molte azioni con molti pulsanti che vanno premuti al momento giusto e in maniera coordinata. Allucinanti le prime esperienze con Defender quando devi giocare anche all'indietro mentre fino a quel momento il plot dei giochi prevedeva o uno schermo fisso o il movimento solo in una direzione.

Si sperimentano nuovi aggeggi di input: dal prevedibile volante per i giochi di guida alla più esoterica trackball presente ad esempio in Missile Command. Missile Command è un gioco semplice nella sua logica: dall'alto scendono dei missili con testata nucleare secondo varie traettorie dirette verso sagome di agglomerati urbani (che starebbero per le città della

terra da difendere). Il gioco consiste nel puntare un cursore davanti al missile in picchiata e dirigervi una specie di contraerea per far sì che la testata minacciosa venga distrutta; c'è da chiedersi che efficacia avrebbe una simile difesa dal momento che il follow-out spianerebbe comunque ogni forma di vita sottostante! Ma tant'è la gente ha preso questo gioco in maniera patriottica al punto che è stato studiato come vero e proprio fenomeno sociale. Questo negli Stati Uniti, mentre da noi credo non sia mai arrivato a tanto; personalmente non lo ricordo come un titolo particolarmente divertente. Fra l'altro l'andamento del gioco era abbastanza irritante, con un continuo incremento nel numero di attacchi provenienti dall'alto senza dei veri livelli: non dava respiro!

Missile Command della Atari (1980). Il primo arcade che sperimentò la trackball come dispositivo di controllo.

Un'altra rivoluzione coinvolgente è stata la nascita di giochi con risoluzione grafica 3D. Il capostipite è Pole Position della Atari, che per il titolo realizza anche un cabinet stile cockpit di guida con tanto di volante, marce, acceleratore e freno, il tutto da guidare seduti, come su una vera F1.

Poco importa se della tua monoposto vedi a mala pena la punta del muso e se gli avversari si avvicinano e allontanano a scatti; l'illusione di correre veramente fra i cordoli bianchi e rossi viene suscitata veramente!

Questo stile "di presenza" rispetto all'ambiente con il quale il giocatore interagisce, diventa uno standard e lo troviamo anche nei titoli moderni come in Need of Speed, ovviamente adeguatamente potenziato.

I giochi per i PC casalinghi non appaiono fino al 1982, quando vengono vendute "come il pane" due macchine costruite principalmente per portare l'esperienza dei giochi arcade nelle case: il Commodore64 e il Sinclair Spectrum.

(...segue...)

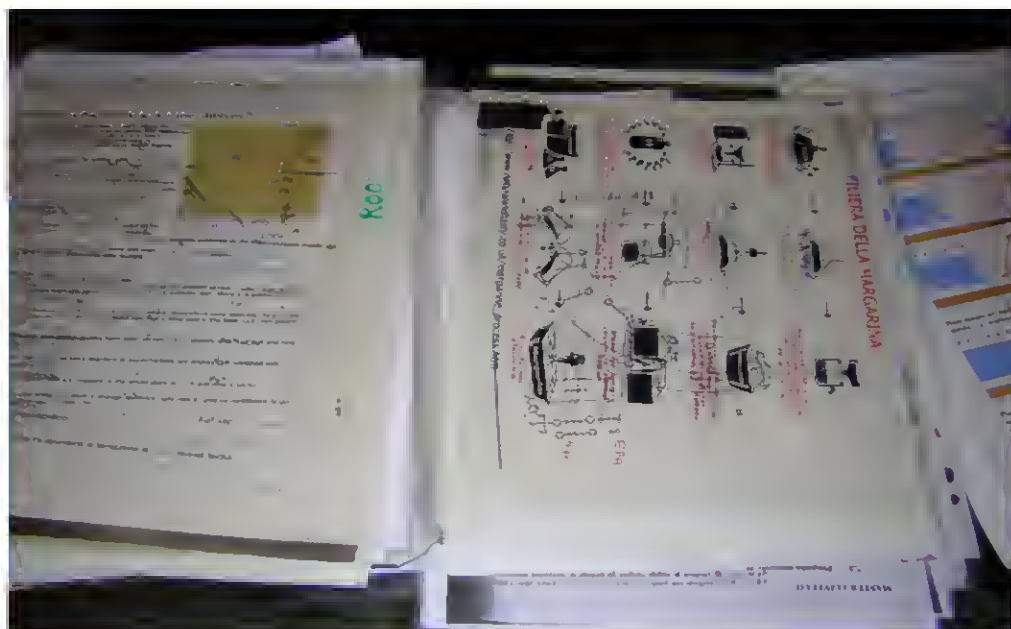
[L2]

Defender, un gioco che introdusse l'innovazione di un controllo spaziale della navicella.

Laboratorio

Interventi hardware e software per la conservazione, il ripristino e l'evoluzione dei sistemi di calcolo personale.

Conservare le fotocopie

Fotocopie e lucidi. Li vogliamo conservare?

Introduzione

Una tipologia di materiale sicuramente molto diffusa, anche fra i retro computeristi, sono le fotocopie. Chi di noi non ha qualche raro manuale fotocopiato da un amico o trovato in dotazione a qualche scheda presa in un mercatino? L'uso dello strumento fotocopia per creare delle parti "agili" di materiale esistente dovrebbe essere il fine ultimo dell'attività di fotocopiatura. Questo significa che si tratta di materiale fatto "al volo" e che si prevede di usare per un tempo limitato, ma ovviamente può trasformarsi in un oggetto di un certo pregio qualora la riproduzione su fotocopia sia l'unica testimonianza esistente di un certo materiale.

Sgombriamo subito il campo delle speranze dicendo che la fotocopia, come materiale atto a tramandare l'informazione, non vale nulla: troppo aleatoria è la sua vita e troppo dipendente dalla qualità della macchina che è servita a riprodurla. Esistono ovviamente sistemi "industriali" che si avvalgono della tecnologia per produrre un prodotto di qualità, ma se ci limitiamo alle nostre per forza di cose limitate esperienze e disponibilità, dobbiamo annoverare la fotocopia fra il materiale meno pregiato che possiamo vantare di possedere.

I fattori che influenzano la durata delle fotocopie sono quelli che si intuiscono: qualità della carta e dell'inchiostro, accuratezza di riproduzione della macchina, espe-

rienza dell'operatore che materialmente ha riprodotto gli originali e la qualità dell'immagine di partenza. Non ci occupiamo in questa sede della misura dell'indice di degrado associato alla copia di varie tipologie di documenti, immagini, stampe, schemi, etc... ma ci concentriamo sull'oggetto fotocopia in sè e per sè, studiano come essa possa essere conservata il più a lungo possibile.

Con questo "il più a lungo possibile" intendiamo il fatto che per le nostre esigenze personali, di documentazione spicciola, è sufficiente che almeno si riesca a leggere, non ha importanza se non passerebbe mai attraverso l'indagine di un OCR: ci basta che sia decifrabile senza troppo sforzo, il suo contenuto.

i cosidetti "lucidi", che si usavano per le lavagne luminose, si possono assimilare a fotocopie, anche perché in una certa misura prodotti con stampanti o proprio fotocopiati.

L'acetato come supporto è migliore della carta, ma ovviamente ha un costo proibitivo per chi volesse riprodurre un libro su questo tipo di materiale, oltre che essere più ingombrante.

In genere questi lucidi durano di più di una semplice fotocopia su carta anche perché normalmente sono conservati con cura: ad esempio si usa tenerli separati da fogli di carta velina, non vengono piegati o messi sopra un termosifone, etc... Insomma quando abbiamo a che fare con questo materiale normalmente lo trattiamo meglio, istintivamente, forse anche perché costava fatica fare un lucido a mano...

Personalmenteho ancora i lucidi che mi servirono per la discussione della mia tesi di laurea: sono ancora perfetti e i colori non sbiaditi (si parla di 25 anni orsono).

Standard internazionali

Dal momento che si tratta di materiale cartaceo. Esiste una vasta letteratura in materia e degli standard ben precisi per le attività di conservazione, così come esistono gli stessi strumenti per la conservazione dei volumi a stampa.

Sulla qualità della carta esiste lo standard Z39.48 (Permanence of paper for Printed Library Materials); per le tecniche di conservazione e immagazzinamento ci si riferisce allo standard ASTM/D3290 (Bond and Ledger Paper for Permanent Records); e infine lo standard ASTM D3458 (Copies from Office Copying Machines for Permanent

Fotocopie che passione! In qualche caso sono ancora indispensabili, come nelle biblioteche universitarie. La macchina delle fotocopie è quasi come la macchinetta del caffé: ci si trova spesso a chiacchierare e ci sono spesso un sacco di ragazze nei paraggi. Occhio però che non vi facciano gli occhi dolci solo per farvelo fare a voi il lavoro!

E'facile ritrovarsi con pile di manuali fotocopiati, soprattutto se fate gli informatici.

Records). Quest'ultimo si riferisce al materiale a stampa, anche a colori, prodotto da stampanti da ufficio o fotocopiatrici non professionali.

Va da sè che è di gran lunga preferibile trasferire le fotocopie in un formato elettronico, incappando sì nei problemi di persistenza di quest'ultimo, ma con una ragionevole probabiità di conservare meglio l'informazione disponibile solo su fotocopia.

Riassumendo le indicazioni più utili di carattere generale:

- il toner deve avere una percentuale di "Carbon Black Pigment", necessario per rendere permanenti le immagini fotocopiate (con toner normale si rischia che parzialmente si stacchi dalla carta con il tempo, magari trasferendosi sul retro del foglio che lo ricopre.

- La fotocopiatrice o la stampante laser che produce la copia deve essere periodicamente controllata e regolata con parametri ottimali di quantità di toner emesso, temperatura e pressione di trasferimento dell'inchiostro alla carta.

- Periodicamente va testata la persistenza del toner sul foglio. Questo è il pericolo più immediato cui va soggetto questo tipo di materiale. Non appena si riscontra una percentuale di distacco significativa è necessario procedere alla produzione di una nuova copia (se possibile). Per materiale particolarmente prezioso si può procedere alla "filmatura" del foglio, cioè a ricoprire l'intero foglio con uno strato di film sottile di plastica. Si tratta di un procedimento molto costoso e oneroso in termini di tempo e che per questi motivi raramente applicato.

- Ovviamente si applicano tutti gli accorgimenti fisico-chimici comuni all'altro materiale cartaceo (umidità, temperatura, etc...).

Esistono poi delle indicazioni biblioteconomiche sulla catalogazione di questo materiale. Esse sono valide all'interno di organizzazioni ufficiali deputate alla conservazione (musei e biblioteche), ma che potrebbero fornire qualche spunto anche per la nostra piccola collezione.

- *Le fotocopie dovrebbero replicare fedelmente il materiale originale compresa la dimensione dello stesso e la posizione degli oggetti. Ad esempio il fronte-retro dell'originale dovrebbe essere replicato esattamente.*
- *Il materiale fotocopiato deve avere dei livelli minimi di qualità che consentano di leggere il testo in maniera completa e utilizzare le immagini eventualmente presenti su di esso. Quante volte abbiamo visto delle fotocopie sulle quali si legge il testo ma la foto si è tragica-*

mente trasformata in una macchia indistinta del tutto inutilizzabile?

- *Il materiale copiato deve essere marcato come copia identificando su di esso in maniera accurata gli estremi dell'originale e la data di effettuazione della copia.*

- *La mancanza di pezzi del documento o la parziale illegibilità di esso deve essere indicata specificando ad esempio "poor quality original" che è l'indicazione internazionale secondo lo standard Z39.48.*

- *Le indicazioni di copyrights devono essere trasferite per intero alla copia.*

- *Deve essere riportata l'indicazione del motivo per cui esiste la copia ed eventualmente la locazione del materiale originale, sia esso posseduto dalla stessa organizzazione o avuto in prestito da un diverso ente.*

- *La produzione di fotocopie da materiale originale può essere in qualche caso l'unica alternativa che permette la conservazione del materiale stesso. In qualche caso si arriva a distruggere l'originale per produrre la copia, ad esempio se il libro si sta deteriorando per effetto della vetustà del supporto cartaceo o per l'attacco ormai irrespingibile di funghi e altri parassiti. Le tecnologie moderne consentono oggigiorno la trasformazione in materiale digitale che però non sempre viene considerato una alternativa unica alla conservazione. In genere i bibliotecari preferiscono averne delle copie "fisiche" piuttosto che affidarsi ciecamente alle nuove tecnologie.*

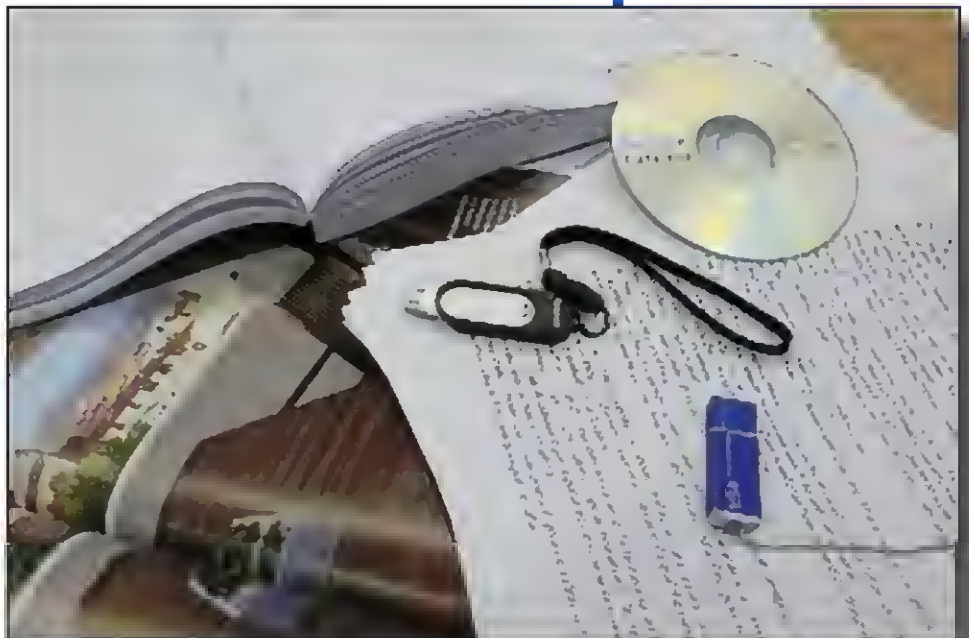
Infine una indicazione di ordine: per quanto possibile le fotocopie andrebbero rilegate. L'esperienza insegna infatti che la rilegatura contribuisce in maniera significativa alla conservazione nel tempo, non fosse altro che per l'effetto di tenere insieme il materiale. Fogli sparsi sono soggetti a perdite, messe fuori ordine e altri guai più deleteri.

Abbiamo visto da queste note, tratte direttamente dalle raccomandazioni del Laboratorio di Conservazione della Library of Congres, quanta cura sia dedicata al materiale di supporto alla conoscenza. Pur con le nuove tecnologie ormai consolidate, anche le umili fotocopie meritano l'attenzione che le sono dovute.

[Tn]

Fotocopie? Meglio trasferirle su CD-ROM...

Il racconto

Rubrica curata da [Bs]

Storie di vita dove i computer (soprattutto retro computer) c'entrano in qualche modo.

Jimmi 'er Fenomeno

Carissimi, vi parlo stavolta di altra gente passata per la megaditta, e di altri tragici eventi accaduti. Vi ricordo che come al solito... è TUTTO VERO! E gli unici eventi non tragici sono... boh? A parte il momento in cui si parla di spettanze, competenze, retribuzione e soldi in generale, non ne ricordo! :-).

In quel tempo era periodo di grandi genialate di programmazione, e facevamo ovviamente a chi la faceva più grossa. Una volta io me ne tirai una talmente galattica (forse fu quando spiegai ar Paolino er secchino, noto torpignattonto residente nella zona di Torpignattara, tutti gli ammennicoli intorno al programma di stampa etichette, dal bypass driver a XXX), che lui, non riuscì a trattenersi dal dirmi: "aó... ma a té te chiamavano proprio Jimmi er fenomeno!" (notare le minuscole di Er e di Fenomeno: voleva riferirsi al personaggio interpretato da Luigi Origene Soffrano). Io, visto che si stava appropinquando il megadirettore ereditario in sala software, e volendo continuare a spararmi le pose, ripetei la frase piazzando le maiuscole e con un accento trionfante: "aó... a mé me chiamavano Jimmi Er Fenomeno!". Proprio in quel momento entravano Arf e il megadirettore megagalattico ed Arf non potè resistere a chiedermi: "e come mai a té te chiamavano Jimmi Er Fenomeno? famme capì, spiegace n'po'". Ed io spiegai, con dovizia assoluta di particolari, e furono grandi momenti di gloria totale (era nato il nuovo personaggio, era finalmente riconosciuto Jimmi Er Fenomeno - ben altro che il vecchio personaggio da filmetti pseudocomici di serie B), il megaboss era muto come un pesce perché l'unica frase plausibile che poteva pronunciare era "ti meriti proprio un aumento, ti meriti un Potente Aumento di Stipendio!", ed infatti, come volevasi dimostrare, NON la pronunciò!

In quei giorni eravamo di rientro dall'ennesima megamissione a XXX nella più mefitica che mitica Volvo, io, il megaboss e Paolino il secchino. Era notte, ed all'altezza del casello di Bologna, mentre azzardava un sorpassaccio, al megaboss venne quasi un colpo: disse: "AH! Idea!" e mentre riusciva per un pelo ad imbroccare l'uscita di Bologna mentre sonori clacson annunciavano notevoli imprecazioni in aramaico ed altre quarantadue lingue da tempo scomparse, pronunciate da altri (e più prudenti)

guidatori sull'autostrada, lui ci spiegava la sua trovata geniale: "Direi di dormire a Modena, conosco un ottimo alberghetto, domattina" [di sabato!?] "ripartiamo per Roma e..." [attimo di pausa] "...vi andrebbe di ripartire col treno stasera stessa?". Paolino annuì di colpo, io non capivo che stava succedendo e -sarà stato il sonno- ho confermato anch'io. Ci prendemmo il nostro bel treno per il rientro, credo verso mezzanotte o giù di lì, da Bologna, mentre il megadirettore ereditario partiva verso Modena. Poi qualche maligno mi disse che lì c'era la sorella di un cliente (quella con XXX XXX) ed altre calunnie simili; io non ho mai potuto verificare e quindi per me rimangono calunnie... ;-)

In quel tempo andavamo spesso, anzi spessissimo a XXX. Ora cor Paolino, ora con Arf, ora con tutti e due, insomma c'erano un mucchio di cose da fare e stavamo sempre in megamissione, ogni settimana, tutti i mercoledì/giovedì/venerdì lì. Il rientro era di venerdì "sera" (cioè il sabato mattina verso le quattro del mattino), ed a seconda degli orari di fine lavoro mi regolavo coi treni da prendere per il rientro all'ovile [a casa]. Di solito, ottimisticamente, pensavamo di partire al massimo alle quattro o alle cinque di pomeriggio da lì: ebbene la volta che ce ne siamo andati davvero PRESTO erano come minimo le 19:30. TUTTI, dico TUTTI i venerdì sera si tirava sempre fino a tardi, per un motivo o per l'altro.

I miei treni preferiti erano:

l'espresso delle 20:35 da Milano Centrale per Reggio Calabria, con fermata in piena notte a XXX;

il diretto delle 04:37 da Roma Tiburtina per XXX.

Capite? il diretto delle quattro e trentasette del mattino! Mai treno aveva orario più assurdo! Le FS mi sono riconoscenti per averlo massicciamente usato... :-)

Una volta a bordo del "quattro e trentasette" eravamo in tre persone: il macchinista, il capotreno ed io! Il capotreno, cinque minuti dopo la partenza, venne da me a consigliarmi di mettermi in una carrozza più indietro perché in quella dov'ero non funzionava il riscaldamento! Se fossimo state le solite trecento persone del diretto delle 12:30, sarebbe stato altrettanto gentile? ;-) Non mi chiese neppure il biglietto: ero l'unico a bordo di quel treno e se avessi voluto inventare una scusa del tipo "ce l'ha mio padre, sta più indietro nel treno, ora vado a chiamarlo", avremmo percorso in lungo e in largo un treno vuoto!

Mi sembra di ricordare di averlo preso almeno una ventina di volte, solo una o due delle quali non era di sabato mattina... di sabato notte, voglio dire!

Forse anche per questo fatto mi merito l'appellativo di "Jimmy 'er Fenomeno"!

[Mm]

Edicola

In edicola o sul Web le riviste che parlano di computer, preferibilmente retro o free

HAX - Hax Are eXperience

Scheda

Titolo: *HAX - Hax Are eXperience*

Editore: LUG Cosenza

Web: *http://hax.cosenzainrete.it*

Lingua: *Italiano*

Prezzo: *Free*

Primo numero: *Maggio 2007*

Periodicità : *Aperiodica*

Di LUG (Linux User Group) ne esistono un po' dovunque in Italia e ognuno di essi probabilmente edita una fanzine o qualcosa del genere che assomiglia ad un bollettino.

Un gruppo nel profondo Sud, a Cosenza per la precisione, tenta la strada di qualcosa di meno precario e rilascia free su Web una rivista vera e propria (o quello che dovrebbe essere un prototipo di una rivista) dedicata al mondo Linux e Free Software in generale.

Il copioso elenco di persone che collaborano all'iniziativa e un discretamente aggiornato sito Web a supporto (http://hax.cosenzainrete.it/hax), dovrebbero essere garanzia di una continuità di uscita del periodico, peraltro "aperiodico" per definizione. Ma a guardare bene il calendario delle uscite, una certa difficoltà a proseguire dopo l'entusiasmo iniziale non è nascosto (speriamo bene). Già troppe di queste iniziative partono a spron battuto per finire poi nelle sabbie mobili delle difficoltà. La pubblicazione di una rivista, seppure on-line, non è cosa banale: i particolari sono numerosissimi dei quali tenere conto e basta che si comincino ad accumulare piccoli ritardi nella consegna dei pezzi che subito si va alle calende greche per la data di uscita.

La rivista dichiara come obiettivo il rilascio cartaceo, cosa che dubito molto possa realizzarsi in mancanza di uno sponsor ben inserito nel mondo dell'editoria. A be guardare non è che il proposito sia

peregrino: alla testata non manca poi molto per diventare una pubblicazione cartacea al pari di quelle già disponibili nelle edicole italiane. (Beh, diciamo che oggi mi sento buono, và).

Al momento in cui scrivo questa recensione ne sono usciti cinque numeri, compreso il numero zero, ma della controparte cartacea nemmeno l'ombra...

Nel perfetto spirito GNU, la testata Hax ospita gli argomenti che sono il core di questa filosofia di utilizzo dei calcolatori personali, il tutto ben scritto e presentato con adeguata grafica e buona impaginazione.

Gli argomenti vanno dalle distribuzioni al resoconto delle iniziative "fieristiche" e "conferenziali" per arrivare ai tutorial dei programmi più utilizzati (OpenOffice.org, The Gimp,...).

Non manca la cronaca dell'ennesima visita in Italia di Stallman e l'intervista/scheda a Linux Torvard (doverosa, ci mancherebbe!).

Ottima cura nella grafica, come dicevo, anche se lo stereotipo del pinguino troneggia su ogni copertina eccetto sul numero zero, dove trova posto una bella ragazza con tanto di portatile e area sognante (un'immagine decisamente preferibile al pupazzetto Tux, per quanto quest'ultimo possa risultare simpatico).

In conclusione una iniziativa da applaudire e supportare ma che dubito molto possa arrivare alla pubblicazione in edicola. Per ora direi che ci si può accontentare di "quello che passa il convento" e godersi queste quaranta pagine, magari non originalissime, ma comunque valide e, cosa che non guasta, senza cacciare il portafoglio!

[Sn]

Biblioteca

Le monografie vecchie e nuove che rappresentano una preziosa risorsa per chi ama il mondo dei computer in generale.

Scheda

Titolo:

The BASIC Conversions Handbook for Apple, TRS-80 and PET Users

Autore:

Brain Bank, david A. Brain, Philip R. Oviate Paul J. A. Paquin, Chandler D. Stone Jr.

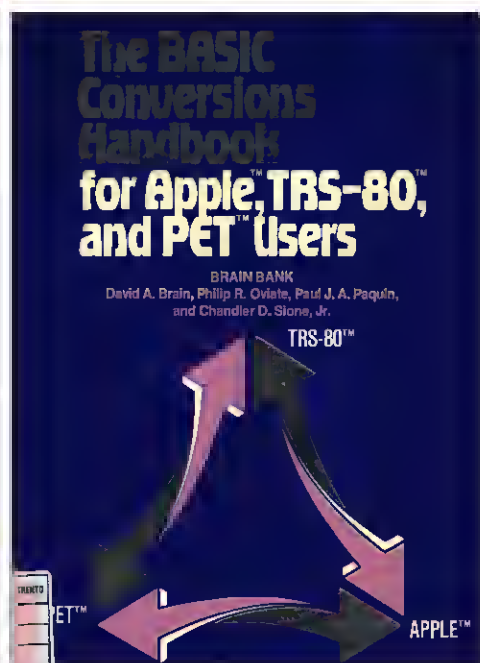
Editore:

Hayden Book Company, inc.

Anno: 1982

Lingua: Inglese

ISBN: 0-8104-5534-X

The BASIC Conversions Handbook

I primi personal computer che hanno avuto una diffusione oltre i semplici appassionati di elettronica sono stati tre e precisamente: il radio Shack TRS-80, la serie PET della Commodore Business Machine (poi solo Commodore) ed infine l'Apple II della Apple Computer.

Questo erano equipaggiati con una propria versione dell'interprete BASIC e questa mancanza di uno standard rigido era ovviamente fonte di problemi.

Se infatti da un certo punto di vista, quello dello sfruttamento ottimale del progetto, la personalizzazione dell'interprete costituisce una arricchimento del prodotto, dall'altra impedisce agli utenti un libero scambio del software, dovendo affrontare il problema della conversione da una versione all'altra del codice.

Bisogna dire che all'epoca di uscita dei primi home, la preoccupazione dei costruttori non era particolarmente pressante sul software: si pensava che fosse sufficiente fornire un buon interprete e poi i programmi l'utente se li avrebbe costruiti da solo. Ovviamente nulla di più sbagliato: l'appetito vien mangiando e, motore la innata flessibilità di questa moderna macchina inventata dall'uomo, gli utenti cominciarono a cercare spasmodicamente di aumentare la propria collezione di programmi.

Dal momento che supporti e collegamenti fra sistemi erano ben lungi dall'essere standard i primi e aldilà dal venire i secondi, l'azione veniva svolta praticamente in maniera esclusiva via tastiera, cioè ognuno si creava/copiava a mano i programmi sul proprio computer.

Sembra incredibile che ciò potesse accadere, nel senso che oggigiorno noi non andremo a scrivere che poche righe di script se ci venissero proposte da qualche rivista, mentre, se proprio non fac-

ciamo della programmazione la nostra professione o il nostro hobby, mai ci passerebbe per il capo di scrivere centinaia di statement per ottenere una certa elaborazione.

Poco meno di una trentina di anni orsono invece questa era la pratica e le riviste di soli listati spopolavano, i magazine si vergognavano di uscire senza lunghe pagine di puro codice e le monografie stesse erano oggetto di questa pratica: titoli come "100 programmi per il tuo xxx" ne sono usciti a centinaia.

I primi utilizzatori di personal si sono quindi trovati difronte a due circostanze negative: il parco macchine dello stesso tipo era poco diffuso e l'impossibilità di scambiarsi il codice con le altre piattaforme a causa proprio dell'incompatibilità degli interpreti.

Il terreno era fertile per una pubblicazione come questa che presentiamo: un manuale di conversione che promette di aiutare coloro che vogliono passare il codice da una piattaforma all'altra. Sembra semplice costruire una simile opera, ma dobbiamo mettere in conto la data di uscita di questo volume: 1981. Allora era assolutamente impensabile disporre ad esempio di tutte e tre le macchine ed averne una conoscenza sufficiente ad affrontare un problema di meta-livello, come direbbero i logici matematici. Ecco spiegato la lunga lista di autori che hanno collaborato alla stesura del testo: ben 5, che per un manualetto di 80 pagine sembrano effettivamente tanti.

Il testo è organizzato in tre parti che contengono le istruzioni per passare il codice fra una delle tre coppie possibili di macchine per un totale di sei direzioni di conversione (da Apple a TRS-80 e ritorno, ad esempio).

Qualora l'istruzione particolare non esista sulla piattaforma di arrivo se ne offrono due alternative: cambiare l'istruzione di partenza oppure costruire una funzione utente che la simuli sulla piattaforma di destinazione.

A questo proposito in una appendice trovano posto le istruzioni per definire questo tipo di funzioni, ad esempio la comoda ed usatissima PRINT USING. Altre tabelle contengono i codici ASCII e il loro utilizzo/significato nei tre personal presi in considerazione.

Conclusione.

Non possiamo pretendere di aggettivare questo testo come "attuale", come del resto è destino comune alle monografie tecniche che hanno solo qualche anno sulle spalle, semplicemente è interessante notare come "ce la cavavamo" nell'epoca in cui tutte le macchine erano diverse, persino quelle prodotte dalla stessa azienda.

[Tn]

Retro Riviste

La rassegna dell'editoria specializzata dai primi anni '80 ad oggi

Bit games

Scheda

Titolo:
Bit
Sottotitolo:
Speciale giochi

Editore:
Jackson
Lingua:
Italiano
Prezzo:
L 2.500

Numero:
Gennaio 1981

Nota: fascicolo monografico della rivista Bit

Giochi, giochi, giochi.... Un grido disperato si alza dall'italico suolo. Abbiamo comprato il "Computer" per casa e dopo averlo mostrato ad amici e parenti suscitandone a seconda dei casi ammirazione o compassione, ora che ci facciamo?

-"Sai, Mi serve per lavoro..." abbiamo mentito: chi mai poteva illudersi di fare qualsivoglia lavoro produttivo su uno ZX80?

-"Mi serve per studiare ..." va già meglio: l'elettronica digitale stà irrompendo e qualche analista già si sbilancia a progettare un roseo futuro per questa nuova branca del consumismo.

Ma infine diciamoci la verità, questo "home" l'abbiamo preso per giocarci. Anche chi si cimenta fra le prime istruzioni GOTO non può fare a meno di incominciare con i giochini: si imparano tante cose e hanno il vantaggio di rendere l'apprendimento meno serioso.

"Perchè Bit Games?" si chiede nell'editoriale Aldo Cavalcoli: si tratta di un fenomeno sociale che stà assumendo proporzioni importanti. Bando agli anatemi scagliati dai sè dicenti "professori illuminati", pronti a prevedere la morte della nostra stessa società, un imbarbarimento dei rapporti interpersonali e la dipendenza fisica all'accoppiata tastiera-TV. Come sempre è importante un sano equilibrio: giocate sul compute sì, ma non dimenticate la partitella a calcetto con gli amici.

In realtà di usabile c'è abbastanza poco e per rendersene conto basta scorrere le pagine di questo numero monografico. Listati e ancora listati, pochi output e nessuna immagine decente che mostri le capacità grafiche dei sistemi

in voga. Siamo alla preistoria del gioco elettronico, epoca giurassica appunto, come abbiamo intitolato questa la nostra rivista.

Evidentemente ci si accontentava e lo scopo principale non era quello di guidare un qualche personaggio in prospetteva 3D in un fantasmagorico scenario alla “Tomb Rider”, ma avere quel tanto di illusione che la macchina risponda in maniera “intelligente” alle nostre mosse.

Ciò nonostante il settore è in crescita. Lo si deduce dalle offerte di giochi per le piattaforme più diffuse e dalla continua uscita di gadget dedicati. Per non parlare delle scacchiere elettroniche, cioè i computer dedicati al gioco strategico per eccellenza, gli scacchi, appunto, ai quali è dedicato un lungo articolo molto interessante che permette di dedurre le capacità di “ragionamento” delle macchine dell’epoca.

Il massimo si raggiunge con il gioco “Gran Prix” (foto a fianco) dove il circuito viene disegnato “graficamente” utilizzando l’interfaccia a caratteri, l’unica disponibile in moltissimi “home”. Certo che per “godere” di questa meraviglia dovete sobbarcarvi la digitazione di circa 300 istruzioni Basic, se riuscite a decifrarle nella riproduzione del listato!

Entrano in campo anche la calcolatrici programmabili. Se avete una TI-58 o TI-59 potete cimentarvi nel gioco “Hovercraft”: 176 passi di programma che vi permetteranno di andare in caccia del nemico su un foglio millimetrato 100x100. E’ come avere un sonar: la calcolatrice vi dà il risultato di una misurazione tramite la quale voi siete in grado di tracciare con il compasso (ma si usa ancora?) un cerchio entro il quale si trova il nemico. Insomma un connubio fra elettronica e geometria euclidea.

La Casio ha prodotto uno strano ibrido fra una calcolatrice programmabile e una tastiera musicale elettronica, la Casio Melody-80. Il primo computer multimediale? Semplicemente si ha la possibilità di pilotare l’autoparlantino incorporato per produrre suoni elementari ma indubbiamente affascinanti per i cultori della musica.

In definitiva un numero monografico particolarmente riuscito che ha la funzione di fotografare lo stato del settore all’epoca di uscita della rivista (1981).

[Sn]

Retro Linguaggi

La storia dell'informatica è stata anche la storia dei linguaggi di programmazione.

ABAP (parte 8)

Realizzeremo in questa lezione del corso la nostra prima tabella e impareremo a gestirla attraverso semplici programmi di tipo REPORT che ormai sappiamo costruire con facilità.

La realizzazione di proprie tabelle, cosidette "tabelle utente", è perfettamente possibile in SAP, utilizzando le funzioni di creazione degli oggetti del repository information system.

partendo dal menù generale e seguendo il percorso dell'ambiente di sviluppo (vedi figura 1), si arriva all'esecuzione della transazione ABAP Dictionary.

Figura 1.

Come si vede dalla figura 2, è possibile creare vari tipi di oggetti: tabelle e viste prima di tutto, ma anche strutture dati ed altri elementi che non esploriamo per questione di semplicità nell'esposizione.

Concentriamoci sulla creazione di una tabella. Scegliamo un nome compatibile con i dettami imposti da SAP (inizio del nome con le lettere Z o Y), ad esempio ZMX_TAB_001 e premiamo il tasto "Create".

Dopo aver inserito una descrizione nel campo "Short Text", come di consueto, scegliamo il tipo di "Delivery Class", come mostrato in figura 3. La scelta più normale è il codice "A", cioè la nostra sarà una tabella "Application Data", contenente dati di applicazione.

Questa informazione riguarda il come la tabella seguirà le regole di trasporto sulle varie istanze di SAP disponibili. Infatti esiste un sistema di "copia mandanti" che effettua la copia dei dati di tutte le tabelle marcate come "A" fra istanze della stessa installazione o anche di installazioni diverse.

Come si vede dalla figura 3, per definire completamente una tabella è necessario specificare dei parametri contenuti in ben cinque sheat: "Attributes", "Delivery and Mantenaince", "Fields", "Entry help/check" e "Currency/Quantity Fields".

I dati "Attributes" (figura 4) si riferiscono all'autore, alla data di creazione e al package cui appartiene la tabella. In prima battuta non è necessario modificare nulla.

Passiamo quindi alla parte più sostanziosa della creazione della tabella, cioè la definizione dei campi.

Diamo prima di tutto uno sguardo ai comandi e ai principi fondamentali della creazione dei campi nella tabella (si veda la figura 5).

E' evidentemente che la prima cosa da scegliere come attributo di un campo è il suo nome. Questi può essere qualsiasi, anche se ci sono delle limitazioni di lunghezza che ne limitano la scelta a 8 caratteri alfanumerici (il primo carattere deve essere alfabetico). Non ci sono limitazioni che obbligano a scegliere nomi che iniziano con Z, ad esempio; i campi sono oggetti interni, come le variabili in un programma.

Notiamo poi dalla figura 5 che per ogni campo, che viene definito su una riga della tabella visualizzata, ci sono delle parti in grigio: queste sono disabilitate, come si usa nella normale interazione uomo.macchina stabilita come standard da Windows.

Per ogni campo si può scegliere di spuntare il checkbox "key" per definire che esso è chiave della tabella e il checkbox "Initial value" se si desidera impostare un valore di default.

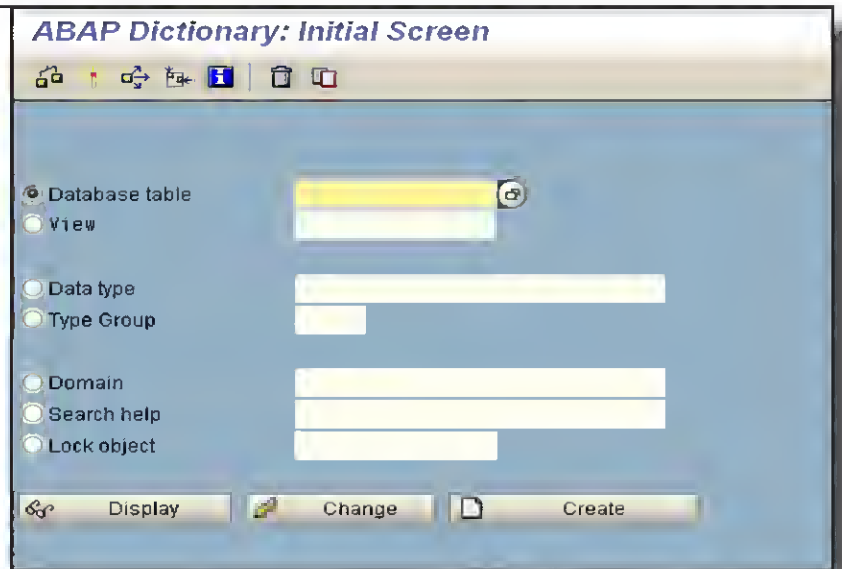



Figura 2.

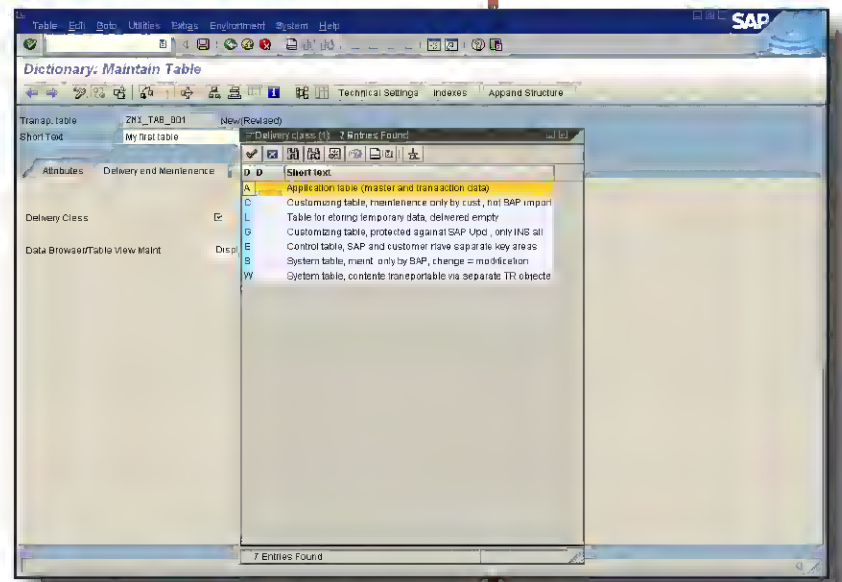



Figura 3.

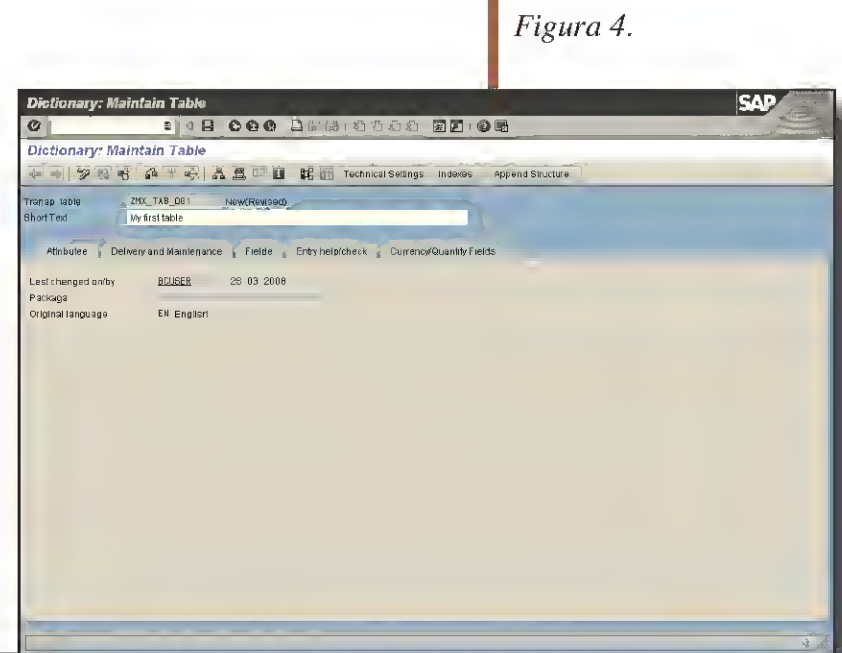



Figura 4.

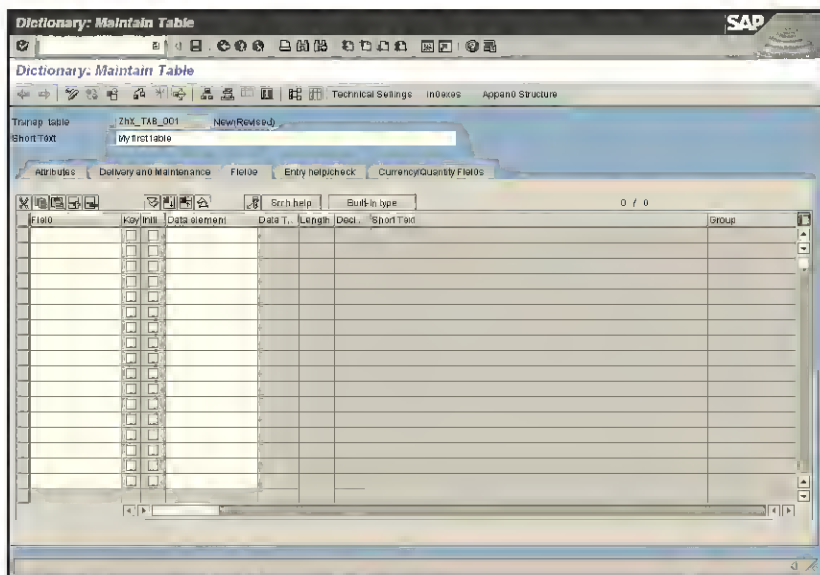



Figura 5.

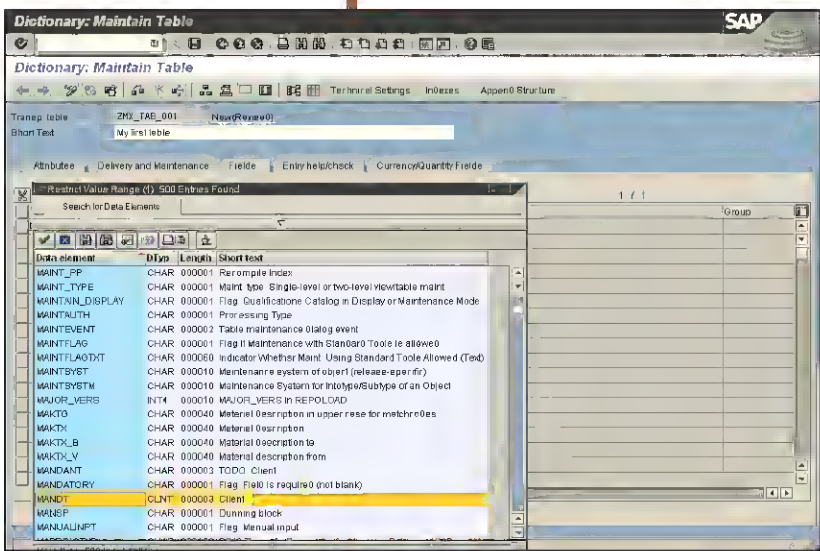
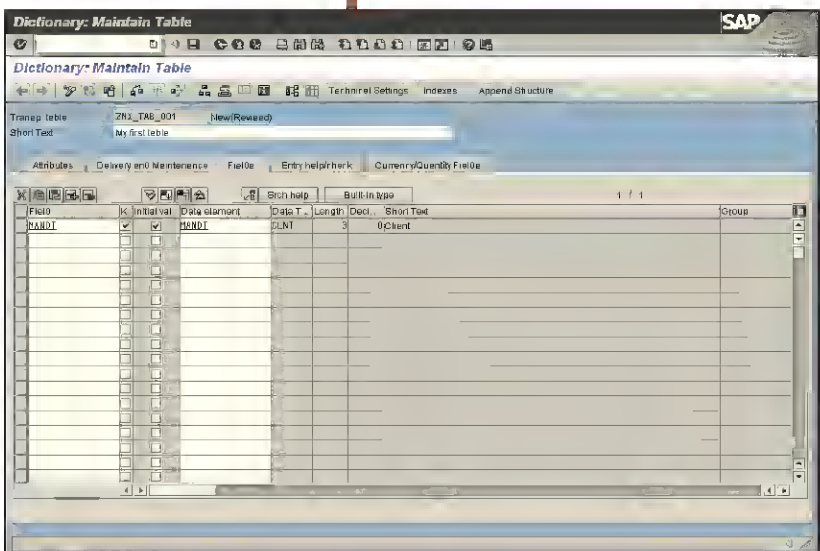



Figura 6.

Figura 7.

Ancora più a destra incontriamo la colonna "Data Element" e qui ci dobbiamo soffermare per introdurre un aspetto fondamentale di SAP: la definizione del tipo di campo.

Questa colonna "Data Element" è alternativa a quelle che seguono: "Data Type", etc...

Per abilitare l'uno o l'altro tipo di definizione si utilizza il bottone più a destra nella toolbar immediatamente superiore alla tabella stessa: "Built-in type". Quando si preme questo bottone esso abilita i campi per definire un tipo di dato "elementare" o built-in, appunto, definendone il tipo, la lunghezza, il numero di decimali ed infine una breve descrizione opzionale.

Se l'utilizzo di tipi elementari è intuitivo e rispecchia più o meno quello che si fa in qualunque programma di gestione di un DB relazionale (Access per esempio), la scelta del tipo "Data Element" è particolare dell'ambiente SAP.

Si tratta di utilizzare come indicazione di tipo un oggetto che è stato definito da qualcun altro (o anche da noi stessi naturalmente) e già presente nel repository.

Un esempio è il campo "mandante", quello che abbiamo visto essere presente in tutte le tabelle SAP che abbiamo finora utilizzato. Questo campo, che si chiama per convenzione MANDT, non lo definiamo come dato elementare built-in di tre caratteri numerici, ma utilizzando la definizione già

predisposta e che ci assicura della scelta corretta in merito a tipo e altri attributi del campo.

Dopo aver predisposto la scelta del tipo in modo da abilitare la colonna "Data Element", inseriamo il nome del campo "MANDT" e definiamo che esso sia una chiave per la tabella, spuntando il checkbox relativo.

Nella colonna "Data Element" è disponibile una finestra di help che ci permette di scegliere il tipo fra quelli già definiti (figura 6).

In questa fase non approfondiamo ulteriormente il come si costruiscano questi "Data Element", cosa che vedremo più avanti nel corso. Ci interessa per ora arrivare alla costruzione di una nostra tabella in modo da utilizzarla nei nostri primi elementari programmi.

Scelto il tipo "giusto", che nel nostro caso è il data element MANDT, il campo è completamente definito sia come attributi fisici: tipo built-in, lunghezza, etc... che come informazione semantica: sappiamo che si tratta del campo che definisce il mandante dell'istanza dove la tabella sarà trasportata ed utilizzata.

Definiamo ora un campo "CODICE" di tipo numerico di cinque caratteri. Il tipo da scelgliere fra quelli proposti nella lista (vedi figura 8) è NUMC. Quindi ne assegnamo la lunghezza (5 digit senza decimali) e mettiamo una descrizione nella colonna "Short Text".

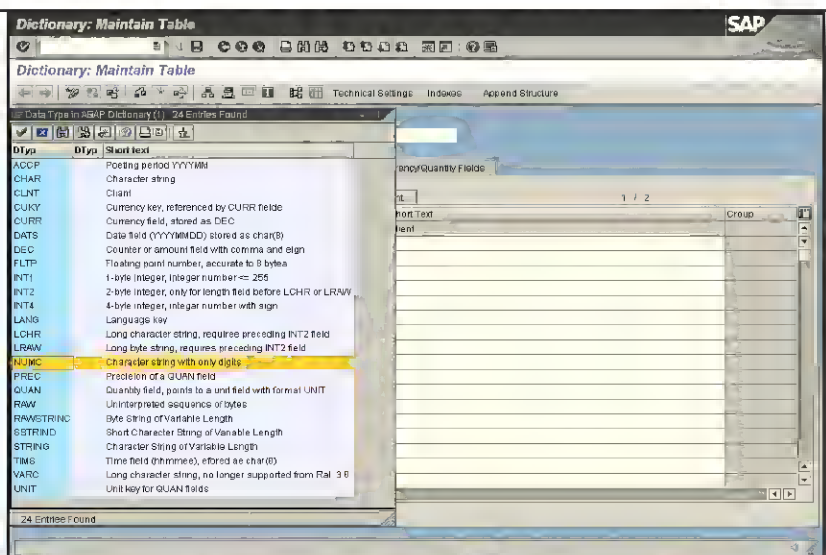



Figura 8.

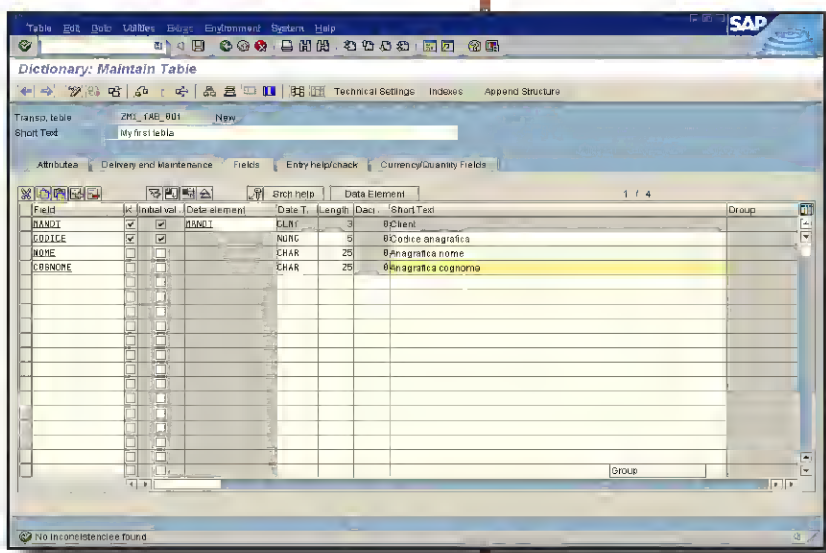



Figura 9.

Ecco nella figura 9 la definizione di tutti i campi della tabella. Abbiamo costruito una semplicissima tabella anagrafica contenente un codice e due campi alfanumerici per il nome e il cognome.

Al momento del salvataggio ci verrà chiesta la solita informazione riguardante la classe di trasporto e il package cui appartiene l'oggetto appena creato; confermiamo "Oggetto locale" per continuare.

Non andiamo ad interessarci dei due sheet aggiuntivi che riguardano le tabelle di riferimento e la valuta.

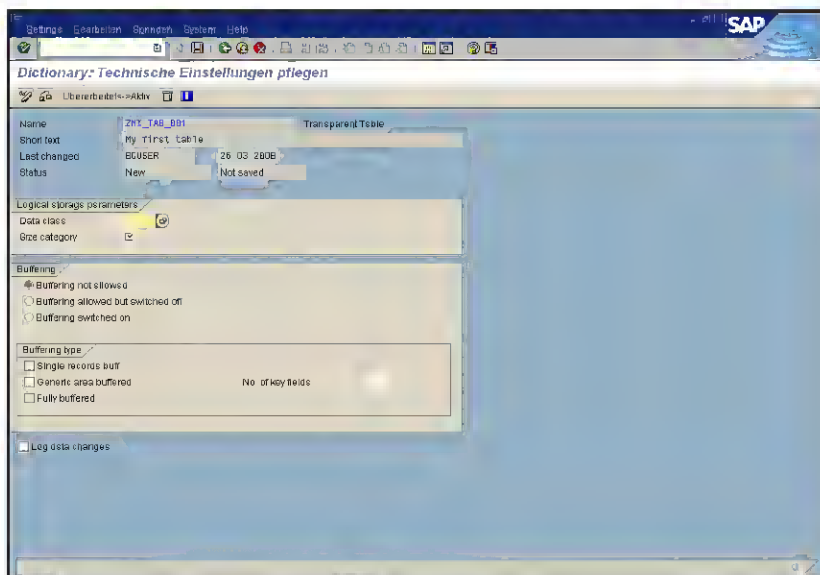



Figura 10.

Prima di usare la tabella dobbiamo ancora passare per la definizione di certi attributi tecnici che si raggiungono con il bottone “Technical Settings” presente sulla toolbar.

Le informazioni tecniche che riguardano la tabella sono piuttosto numerose (figura 10), ma noi inseriremo per ora solamente le due essenziali che sono: “Data Class”, da impostare al valore “APPL0” e “Size category” che impostiamo a “0” (non prevediamo più di 10.000 record per la nostra tabella, vi pare? :-)

A questo punto il nostro oggetto tabella è pronto, ma per essere usato in un programma deve essere attivato, che significa reso disponibile nell’ambiente di sviluppo e all’utilizzo del mandante corrente.

Dopo l’attivazione (che è anche un momento di check dei dati inseriti), realizziamo un semplice programma “report” per caricare dei record nella tabella.

Il listato 1 è un esempio minimale di inserimento di due record. Il meccanismo è dei più semplici: dopo aver caricato i valori in un’area di lavoro che si chiama come la tabella (ZMX_TAB_001 nel nostro caso), l’inserimento del record avviene con la semplice istruzione INSERT.

Per visualizzare il contenuto della tabella è utile il tool “Contents”, disponibile quando si visualizza l’oggetto all’interno del repository.

Questo tool chiede inizialmente se si vogliono impostare dei range di ricerca di particolari valori (figura 11) e successivamente visualizza il contenuto della tabella secondo i criteri scelti (figura 12).

Se si fa girare il programma più di una volta non si ottengono segnalazioni o inserimenti aggiuntivi per il semplice fatto che l’inserimento di un record con chiave già esistente fallisce.

Come si controlla l’avvenuto inserimento o la presenza di un errore

Listato 1.

```
REPORT ZMX_08_01

TABLES: ZMX_TAB_001.

START-OF-SELECTION.

 MOVE '00001'  TO ZMX_TAB_001-CODICE.
 MOVE 'MARIO'  TO ZMX_TAB_001-NOME.
 MOVE 'ROSSI'  TO ZMX_TAB_001-COGNOME.

 INSERT ZMX_TAB_001.

 MOVE '00002'  TO ZMX_TAB_001-CODICE.
 MOVE 'MARIA'  TO ZMX_TAB_001-NOME.
 MOVE 'BIANCHI' TO ZMX_TAB_001-COGNOME.

 INSERT ZMX_TAB_001.

END-OF-SELECTION.
```

nell'operazione?

ABAP dispone di una variabile di sistema chiamata SY-SUBRC (che starebbe per SUBroutine Return Code) che viene impostato ad ogni operazione con il corrispondente risultato: zero se tutto è andato bene e maggiore di zero se l'operazione non è andata a buon fine.

Vediamo nel listato 2 l'utilizzo di questa variabile di ambiente per controllare l'operazione fatta. Si tratta di una semplice IF per testare il valore della variabile al termine dell'operazione di inserimento.

Analogamente semplici sono le operazioni di cancellazione (DELETE) e di modifica (UPDATE) di un record presente. La tecnica è sempre la stessa: si prepara l'area di lavoro dichiarata con la direttiva TABLES e si esegue l'operazione desiderata. Ad esempio per cancellare un record è sufficiente caricare il valore del codice e poi chiamare la funzione DELETE.

Il tutto con semplicità ed immediatezza che pochi altri liguaggi possono vantare!

Anche la lezione numero otto si è conclusa e anche questa volta abbiamo fatto un passo importante nella comprensione del linguaggio ABAP e nei meccanismi di gestione dell'ambiente SAP. Nella prossima lezione cominceremo a parlare di screen e interfaccia utente, cominceremo cioè la seconda parte del corso.

[Mx]

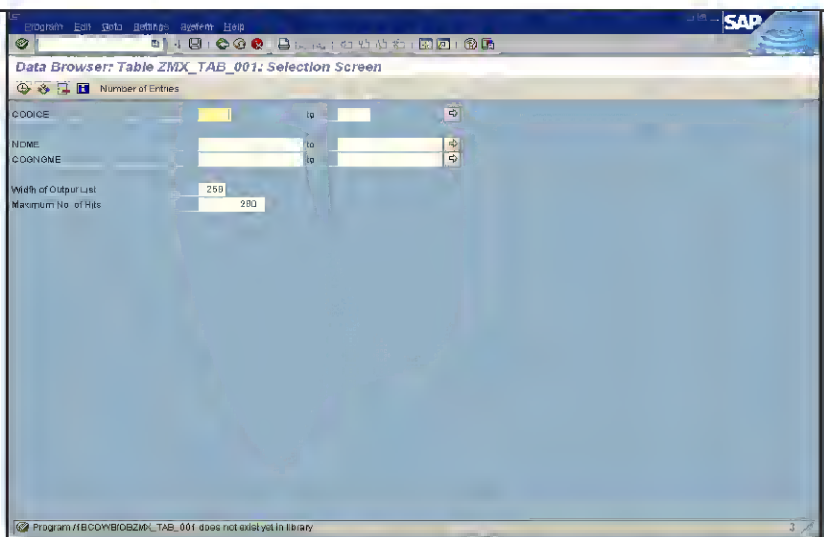



Figura 11 (sopra) e

Listato 2.

```
REPORT  ZMX_08_01                    .

TABLES: ZMX_TAB_001.

START-OF-SELECTION.

 MOVE '00001'    TO ZMX_TAB_001-CODICE.
 MOVE 'MARIO'    TO ZMX_TAB_001-NOME.
 MOVE 'ROSSI'    TO ZMX_TAB_001-COGNOME.

 INSERT ZMX_TAB_001.
 IF SY-SUBRC > 0.
        WRITE 'Impossibile inserire il record".
 ENDIF.

END-OF-SELECTION.
```

DIR

Le facce del CP/M su Apple

Le pagine dedicate al sistema operativo CP/M

Un nuovo mondo si apre agli utilizzatori dell'Apple e Microsoft ne è il profeta...

Introduzione

Vogliamo in questo articolo discutere delle differenze esistenti fra le soluzioni hardware/software che abilitano il sistema Apple II di Apple Computer, all'uso del sistema operativo CP/M.

Come è noto si tratta di un add-on

davvero prezioso per gli utilizzatori di questa piattaforma che possono attraverso esso godere dei vantaggi di un sistema stabile, diffuso e ricco di applicazioni pratiche, fra le quali molti linguaggi di programmazione.

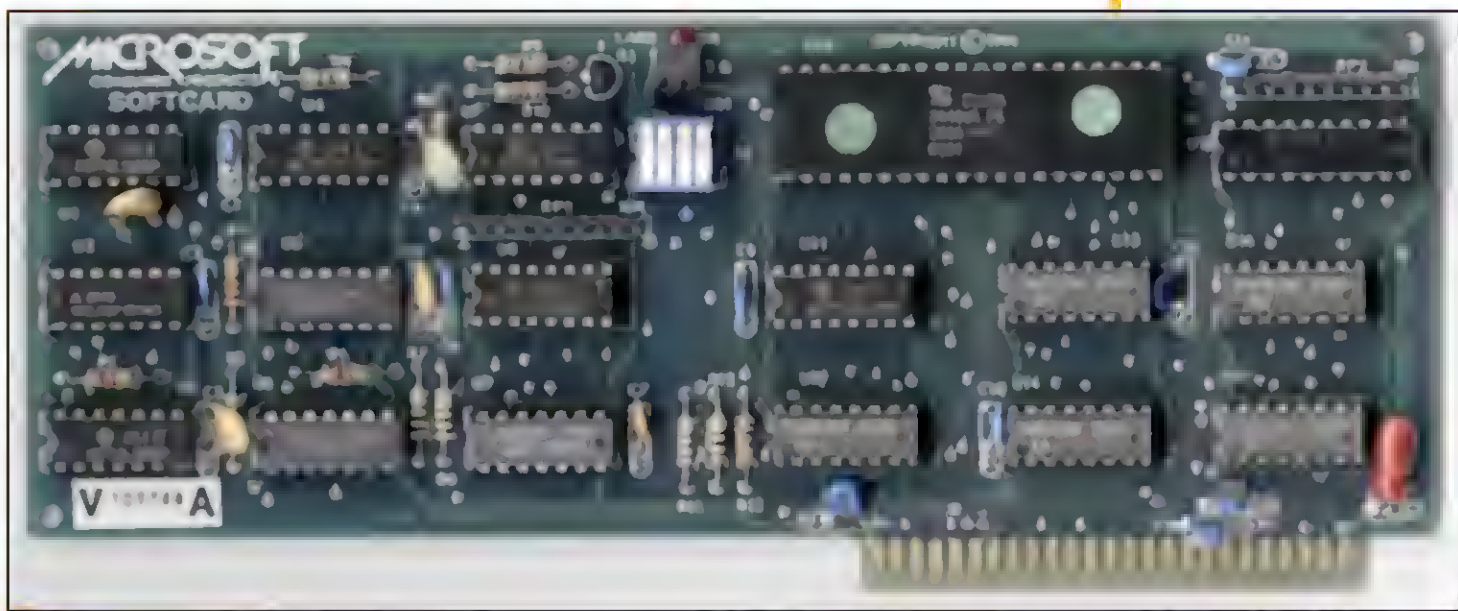
La soluzione "principe" è quella commercializzata dalla Microsoft, a partire dal 1979, completa di scheda di espansione con CPU Z80 e software adattato, come il suo famoso BASIC-80 (o MBASIC che è lo stesso software chiamato diversamente) e compilatore CBASIC. Si dice che la Microsoft ne abbia vendute più di 100.000 fino al 1984, il che non è stato affatto un cattivo affare per la ditta di Bill Gates.

Tralasciando i semplici cloni, semplici copie di progetti esistenti, sistono altre soluzioni in grado di offritre qualcosa di alternativo, anche dal punto di vista del prezzo, ma soprattutto sul fronte della completezza della soluzione. Ne esaminiamo in dettaglio pro e contro delle scelte che possiamo considerare "maggiori" in quanto originali e supportate dal punto di vista tecnico e commerciale.

La soluzione Microsoft (CP/M Softcard) è ovviamente il termine di paragone. Essa offre una buona compatibilità con le periferiche Apple esistenti all'epoca e punta molto sulla compatibilità del software rispetto alla libreria CP/M "originale". Il software per CP/M veniva prodotto per buona parte in un formato per floppy da 8" secondo uno standard IBM, l'uso di questo software è compatibile con la soluzione Microsoft.

Il software particolare che si aggiunge allo standard si compone di utility molto utili (scusate il gioco di parole) per qualche configurazione particolare (ad esempio per un sistema con un solo drive).

APDOS: copia file di testo e binari da un floppy Apple DOS ad uno CP/M;

La scheda Softcard originale di Microsoft

BOOT: esce dal CP/M e rientra nel sistema operativo Apple;

CAT: sostituisce il comando standard DIR del CP/M per listare il contenuto di un supporto aggiungendovi la dimensione dei files e lo spazio libero disponibile;

CONFIGIO: programma di configurazione, da eseguire per customizzare il sistema alle proprie esigenze;

MFT: copia file da un disco all'altro su un sistema che ha un solo drive disponibile.

La Microsoft rilascia la Softcard con software applicativo allegato: MBASIC e GBASIC sono le versioni standard e grafica dell'interprete BASIC-80, adattato alla soluzione per Apple.

Dopo l'entrata nel mercato dei prodotti della concorrenza, che offrivano caratteristiche aggiuntive, magari a minor prezzo, la Microsoft si è precipitata ad aggiornare la propria offerta con una versione chiamata "Premium Softcard IIe System". Tale aggiornamento aggiunge RAM a bordo, portando il totale del sistema a 128 Kb e una CPU a 6 MHz. Permette l'utilizzo della memoria aggiuntiva da Apple DOS come disk RAM e aggiunge una interessante features di bufferizzazione della tastiera e dell'uscita LPT (la stampante). In generale questo nuovo prodotto si pone in diretta concorrenza, anche nel prezzo (circa 300$) con la soluzione APPLI-CARD, adottata anche dalla stessa Digital Research con la sua personale soluzione chiamata STAR-CARD.

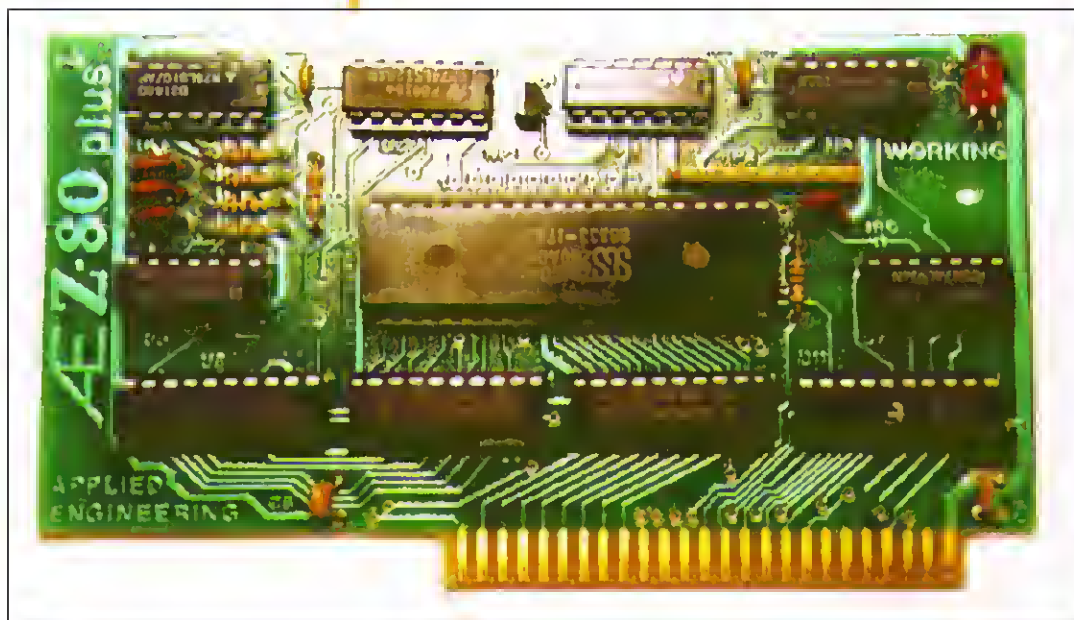
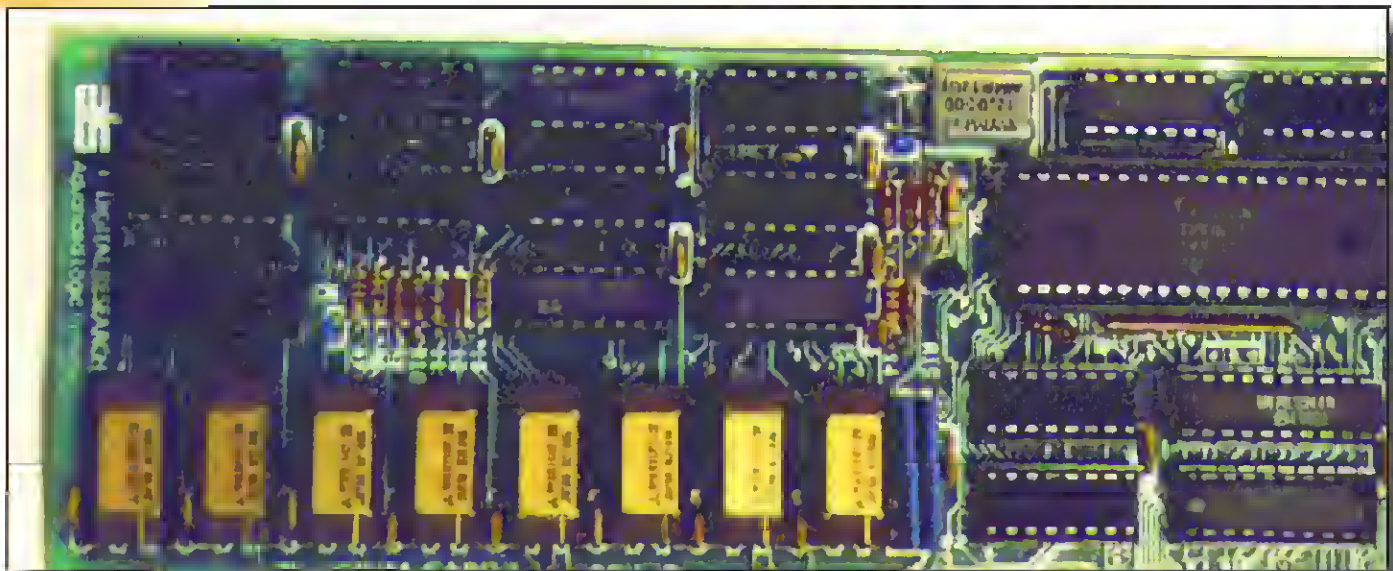
APPLI-CARD/STAR-CARD

E' la soluzione al CP/M offerta da una ditta che si chiama Personal Computer Products che vende la soluzione stand-alone completa di sistema operativo. Lo stesso hardware viene "vestito" dalla Micro-Pro, produttrice del famoso WordStar, per offrire una soluzione "chiavi in mano" che permetta ai possessori dell'Apple l'utilizzo delle soluzioni office della MicroPro.

La scheda "nuda" costa 275$, mentre con Wordstar o Infostar il prezzo sale a 495$.

La Personal Computer Products riesce a render appetibile la scelta

La scheda Z80Plus di Applied Engegnering

La CPM Card della Digital Research e associate

della sua soluzione, dotandola di una CPU Z80 di tipo B, capace di "tirare" fino a 6 MHz, mentre la soluzione Microsoft è ferma a 2 MHz. Il guadagno teorico di 3:1 sembra effettivamente appetibile rispetto alla Softcard che viene venduta allo stesso prezzo.

La soluzione APPLI-CARD (PCPI sui manuali) ha ulteriori tre vantaggi:

- viene venduta con 64 Kb di RAM on-board. In questa maniera si dispone di un sistema CP/M alla sua massima espansione senza bisogno di acquistare la scheda 16Kb nativa del sistema Apple (l'Apple] [veniva venduto in configurazione standard con 48 Kb).

- E' possibile acquistare una ulteriore espansione che porta la RAM a 128 Kb, da usare in parte come soft-disk, anche dall'Apple DOS.

- La PCPI ha la possibilità di simulare una uscita video a 70 colonne, senza possedere hardware aggiuntivo. Questa soluzione è stata studiata ad-hoc proprio per offrire agli acquirenti del word processor la capacità di usare al meglio tale software.

L'aspetto negativo della soluzione PCPI è una certa difficoltà di configurazione che fra l'altro prevede che l'installazione della scheda sia fatta obbligatoriamente nello slot numero 4 dell'Apple, almeno prima di eseguire la configurazione personalizzata. Questo è visto come una certa preoccupazione dai redattori dei prodotti sulle riviste, giudicando difficile per l'utente target lo smanettare all'interno della macchina per aggiungere/spostare schede di espansione.

La lista delle utility personalizzate prevede:

- COPYFRMT: crea una copia del disco di sistema, contenente le eventuali personalizzazioni;

- ADOSXFER: programma di trasferimento fra il CP/M e il DOS Apple 3.3;

- DOSRDSK: drive per il RAM-disk da usare con il DOS Apple che vedrà la RAM come terzo floppy;

CONFIGSV: programma di configurazione.

Advanced Logic System Z-Card II

La ALS Z-Card II è una soluzione economica per dotare il proprio sistema Apple][della capacità del CP/M. Viene venduta a 169$ ma prevede comunque una licenza ufficiale per il CP/M in versione chiamata "Plain Vanilla". A fronte del risparmio c'è da pagare lo scotto di possedere sia l'espansione da 16 Kb di RAM e la scheda a 80 colonne, oltre a non avere la garanzia del funzionamento di tutti i programmi customizzati per girare sulla Soft-Card di Microsoft.

La lista del software aggiuntivo disponibile sul supporto ufficiale è:

- ADUMP: un programma per il dump esadecimale di un eseguibile;

APPMAKER: crea un disco "applicativo" contenente il sistema operativo, il programma applicativo e la customizzazione del file HELLO che lancia l'applicazione al boot;

- CAT: lista alfabetica alternativa al comando interno DIR del CP/M;

- DISCOPY: copia di floppy, compreso il sistema operativo stesso;

- HELLO: è praticamente necessario ad ogni boot; predispone il funzionamento dlela scheda e fornisce un menù di lancio per la scelta dell'ambiente di lavoro al boot;

- HELP: semplicemente lista a video il contenuto di un file contenente le ultime modifiche al manuale;

- MON65: destinato ai programmatori più esperti, consente l'utilizzo del Monitor del sistema Apple per il debug (il manuale dice esplicitamente che deve essere usato a proprio rischio e pericolo);

- WSMAKER: permette di customizzare il programma Wordstar , non fornito con la scheda, per essere usato con la Z-Card II.

ALS "The CP/M Card"

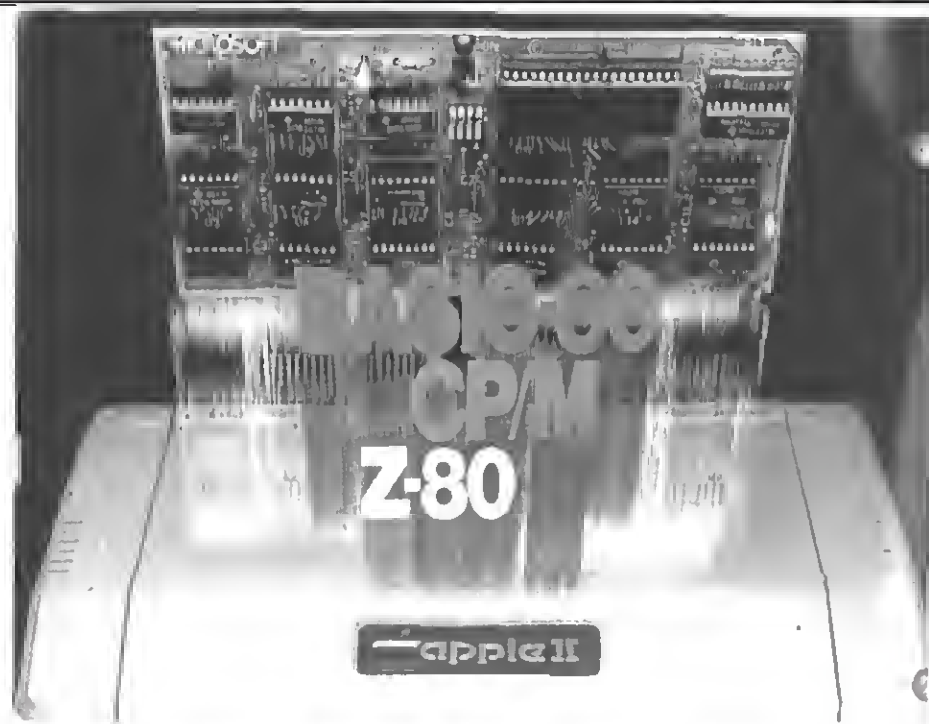
La ALS ha chiamato semplicemente "The CP/M Card" la sua soluzione hardware/software finale , intendendo che ritiene essa la "soluzione finale" per questo settore di mercato. Sviluppata con la collaborazione della Digital Research, proprietaria del marchio CP/M, se ne avvantaggia equipaggiando la soluzione con la più recente versione del sistema operativo, il CP/M 3.0, chiamato anche CP/M Plus.

L'utilizzo della versione 3.0 del sistema operativo non inficia la possibilità di far eseguire sul sistema tutto il software sviluppato per la versione 2.2, lasciando aperta la possibilità per il software di nuova generazione, che fra l'altro si avvantaggia della eventuale presenza di memoria RAM aggiuntiva (fino a 128 Kb).

La qualità però si paga: circa

400$, ma se si considera che il CP/M Plus costerebbe quasi 300$ di listino e il compilatore CBASIC altrettanti, si deduce che il boundle è alla fine conveniente al massimo.

La scheda comprende 64Kb di RAM, non utilizzabile come ram-disk dall'Apple, e può simulare la presenza della scheda a 80 colonne; inoltre la CPU utilizzata è la versione a 6 MHz, ed è perciò al top delle prestazioni.

Il software personalizzato è ridotto all'essenziale e comprende in pratica le stesse utility della scheda Z-Card II della Advanced Logic System.

Turn your Apple into the world's most versatile personal computer.

The SoftCard™ Solution. SoftCard turns your Apple into two computers. A Z-80 and a 6502. By adding a Z-80 microprocessor and CP/M to your Apple, SoftCard turns your Apple into a CP/M based machine. That means you can access the single largest body of microcomputer software in existence. Two computers in one. And, the advantages of both.

Plug and go. The SoftCard system starts with a Z-80 based circuit card. Just plug it into any slot (except 0) of your Apple. No modifications required. SoftCard supports most of your Apple peripherals, and, in 6502-mode, your Apple is still your Apple.

CP/M for your Apple. You get CP/M on disk with the SoftCard package. It's a powerful and simple-to-use operating system. It supports more software than any other microcomputer operating system. And that's the key to the versatility of the SoftCard/Apple.

BASIC included. A powerful tool. BASIC-80 is included in the SoftCard package. Running under CP/M, ANSI Standard BASIC-80 is the most powerful microcomputer BASIC available. It includes extensive disk I/O statements, error trapping, integer variables, 16-digit precision, extensive EDIT commands and string functions, high and low-res Apple graphics, PRINT USING, CHAIN and COMMON, plus many additional commands. And, it's a BASIC you can compile with Microsoft's BASIC Compiler.

More languages. With SoftCard and CP/M, you can add Microsoft's ANSI Standard COBOL, and FORTRAN, or Basic Compiler and Assembly Language Development System. All, more powerful tools for your Apple.

Seeing is believing. See the SoftCard in operation at your Microsoft or Apple dealer. We think you'll agree that the SoftCard turns your Apple into the world's most versatile personal computer.

Complete information? It's at your dealer's now. Or, we'll send it to you and include a dealer list. Write us. Call us.

SoftCard is a trademark of Microsoft. Apple II and Apple II Plus are registered trademarks of Apple Computer. Z-80 is a registered trademark of Zilog, Inc. CP/M is a registered trademark of Digital Research, Inc.

Microsoft Consumer Products, 400 108th Ave. N.E., Bellevue, WA 98004, (206) 454-1315

La soluzione ALS è sicuramente il punto di arrivo per un utilizzatore del software CP/M sul sistema della mela: a fronte di un esborso superiore alle soluzioni concorrenti offre l'indubbio vantaggio di poter disporre dell'ultima versione del sistema operativo della Digital Research.

[Tn]

Bibliografia:

Steven Frankel, The Complete Apple CP/M, Reston Computer Group Book, 0-8359-0800-3, USA 1984

Apple2Info.net

TAMC

Teoria e Applicazioni delle Macchine Calcolatrici: la matematica e l'informatica, le formule e gli algoritmi, la completezza e la computabilità, le strutture dati e tutto quello che sta alla base dell'informatica.

Algoritmi di SORT (parte 3)

Il Quick Sort

Il Quick sort è l'algoritmo di ordinamento più famoso per via della sua efficienza che è superiore a qualsiasi altro; efficienza che si rivela soprattutto quando la dimensione del problema cresce. La sua implementazione infatti implica un certo lavoro, come vedremo, che pesa infinitesimamente nel caso di N grande, ma sarebbe inaccettabile nel caso di vettori di pochi elementi.

L'idea di base dell'algoritmo fa uso del principio divide et impera, che consiste nella pratica nel dividere il vettore da ordinare in sotto insiemi sempre più piccoli, quindi facili da mettere in ordine, e successivamente unirli per ottenere il risultato cercato.

Consideriamo come esempio il vettore seguente:

```
[3, 5, 7, 1, 4, 6, 2]
```

il primo passo dell'algoritmo consiste nello scegliere un elemento del vettore da usare come elemento "pivot". Può essere qualsiasi elemento, ma l'algoritmo funziona meglio se l'elemento scelto stà a metà circa del range di valori da ordinare.

Scegliendo ad esempio il valore 5 il secondo passo consiste nel dividere gli elementi del vettore originale in due sottoinsiemi: quelli inferiori al pivot e quelli uguali o superiori:

```
[3, 2, 4, 1]
```

è il primo sotto-vettore e

```
[7, 6, 5]
```

è l'insieme complementare.

L'iterazione prosegue considerando separatamente i due sottoinsiemi. Dal primo scegliamo il 3 come Pivot ottenendo: [1, 2] e [4, 3]. Il lavoro prosegue dividendo ancora e [1, 2] diventa [1] e [2]. Siamo arrivati alla fine: un insieme di un solo elemento è per definizione ordinato, non vi pare?

Ora scatta la fase di ricostruzione che, in maniera ricorsiva raccoglie questi mini-insiemi di un solo elemento ottenendone il vettore ordinato, la soluzione che andavamo cercando.

Visivamente è utile seguire lo sviluppo delle operazioni guardando l'albero della figura 1. Si capisce bene perché alla fine si arrivi al risultato: il segreto è quello di continuare a dividere esplorando sempre per primo l'albero a sinistra, quello con gli elementi di valore inferiore. Al momento della ricostruzione basta tornare indietro raccogliendo via via tutti gli elementi che a questo punto saranno automaticamente ordinati.

L'algoritmo funziona perché applica un principio fondamentale nella teoria del Problem Solving: la decomposizione del problema in parti più piccole e di conseguenza più affrontabili.

Passare dalle parole ai fatti può non essere immediato. Infatti come ci organizziamo? Usiamo dei vettori di appoggio per tenerci i dati via via raccolti? In realtà il QuickSort è un esempio classico di come si possa usare proficuamente l'area di stack per mantenere i dati temporanei, quindi senza dichiarare memoria in maniera esplicita. Come tutti sanno, e se non lo sanno lo annunciamo ora, il metodo migliore per usare lo stack è quello di utilizzare le chiamate a funzioni con parametri. I parametri vengono infatti conservati in questa area di memoria da dove la funzione stessa li recupera per l'elaborazione. La ricorsione è il metodo pratico con il quale si implementa l'algoritmo, cioè la chiamata della funzione all'interno di se stessa. Nel caso in esame la funzione Quicksort accetterà tre parametri di input: un vettore e due indici che rappresentano i limiti del range di valori da considerare nella sequenza. La chiamata ricorsiva sarà effettuata su una opportuna partizione del vettore di ingresso e così via fino alla conclusione.

Figura 1. L'albero di risoluzione per il vettore di esempio.

Vediamo prima il codice delle due funzioni di appoggio.

La funzione FindPivot (Listato 1) ha l'unico scopo di scegliere un valore dal vettore in modo da garantire che il vettore stesso sia poi divisibile in due sottoinsiemi. L'ideale sarebbe una scelta che individuasse il valore centrale, ma l'algoritmo sarebbe penalizzante e in fondo non è nemmeno necessario che l'albero dei sottoinsiemi che si svilupperà da questa scelta sia perfettamente bilanciato.

Per ragioni "storiche" si usa di solito il codice

Listato 1. La funzione che sceglie l'elemento pivot.

```
Function FindPivot( V, a, b )
        : integer;

for k:=a+1 to b do begin
        if V[k] > V[a] then begin
                FindPivot := V[k];
                break;
        end
        Elseif V[k] < V[a] then begin
                FindPivot := V[a];
                break;
        end;
        FindPivot := -1;
end;
```

```
Function partiziona(V, a , b, pivot)
      : integer;

i : = a; j := b;
repeat
 While V[j] >= Pivot
      j := j - 1;
      While V[i] < Pivot do
            i := i + 1;
            if i<j then
                  Scambia(V[i], V[j]);
until i < j
  Partiziona := j;
end;
```

Listato 2. La funzione di partizione divide il vettore in due parti usando l'elemento pivot come spartiacque.

che è stato mostrato: si tratta di scegliere l'elemento più grande fra una coppia di elementi contigui. Ci si deve assicurare infine che gli elementi del sottovettore considerato non siano tutti uguali: se succede il sotto-vettore è già ordinato. Nel nostro codice questa condizione provoca la restituzione del valore convenzionale -1 da parte della FindPivot.

La seconda funzione è quella che divide il sottoinsieme considerato in due sottoinsiemi:

Partiziona(V, a, b, p), dove p è l'elemento pivot precedentemente trovato con la relativa funzione (Listato 2).

La funzione partizione restituisce semplicemente un indice che è l'indice del primo elemento della parte sinistra del sotto-vettore, il quale contiene tutti i valori maggiori o uguali al Pivot.

L'algoritmo è ora di semplice implementazione e consiste nella sequenza ricorsiva di chiamate alle due funzioni realizzate in precedenza (Listato 3).

```
QUICKSORT( V, a , b )

begin
   pivot := FINDPIVOT( V, a , b )
   if pivot <> -1 then begin
      q = PARTITION( V, a , b, pivot)
      QUICKSORT( V, a , q )
      QUICKSORT( V, q + 1, b )
   end;
end;
```

Listato 3. L'implementazione finale dell'algoritmo di quicksort, nel consueto linguaggio "pascal" che utilizziamo per la realizzazione.

La procedura QuickSort richiama se stessa prima sul sotto-vettore di sinistra e poi su quello di destra. Quando la chiamata alla funzione FindPivot restituisce -1 la procedura è conclusa e quella parte di sotto-vettore è ordinata.

Ad onta dell'apparente complessità dell'algoritmo, esso viene eseguito in meno tempo rispetto a qualsiasi altro. La sua dimensione computazionale è O(2n) che deriva dalla linearità delle due funzioni FindPivot e Partiziona, entrambe di ordine O(n).

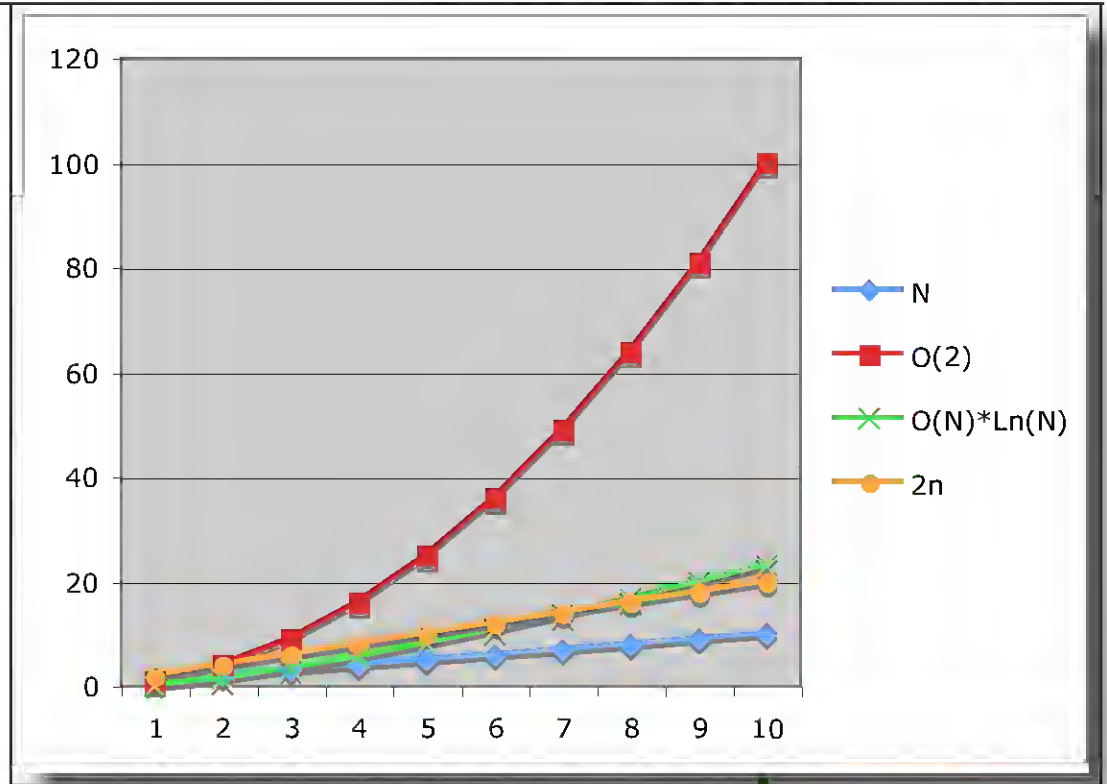
Questo significa che raddoppiano il numero di elementi l'algoritmo utilizzerà il doppio del tempo, sempre considerando valori medi.

Le controindicazioni riguardano l'utilizzo della ricorsione come "arma" per iterare l'applicazione delle funzioni che formano l'algoritmo stesso. Non è detto infatti che non si incappi in un errore di run-time quando lo stack dell'ambiente di esecuzione venga "sfondato" dalla sequenza di chiamate. Il pericolo è tuttavia abbastanza remoto negli ambienti moderni, ma certo non può essere ignorato. Per superarlo esistono due strade:

- dividere il vettore originale in parti "trattabili", ordinarle e poi effettuarne il merge;

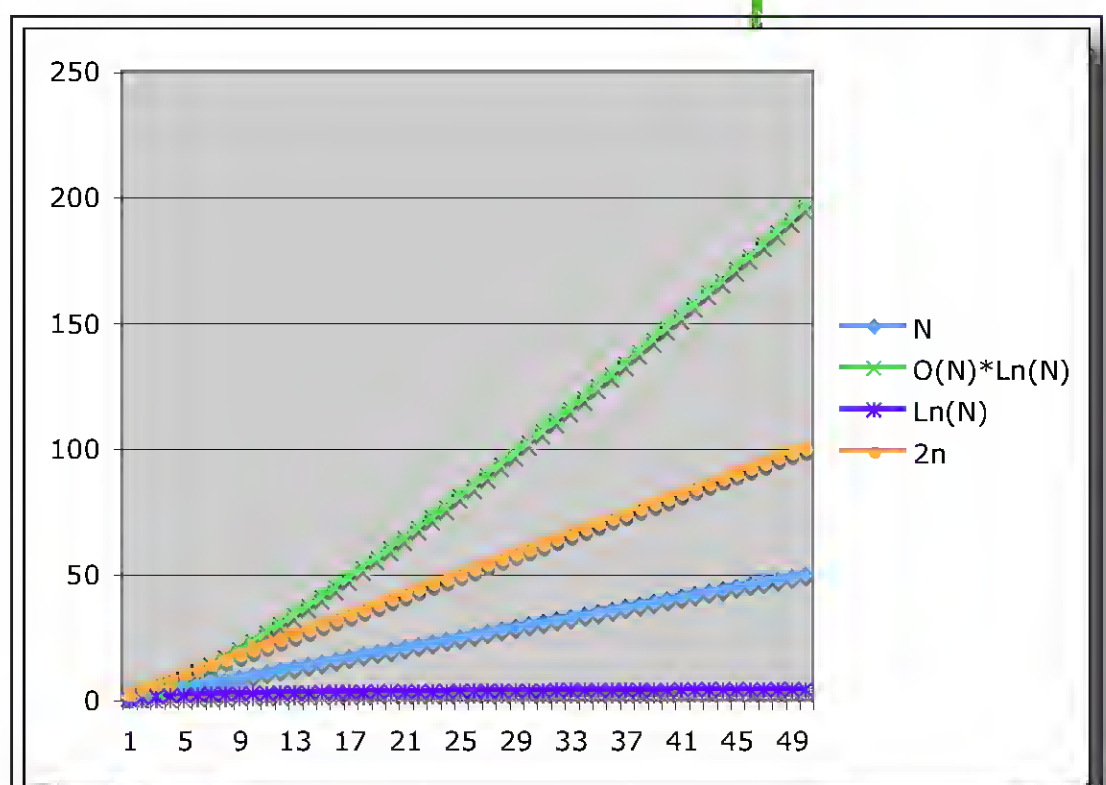
- convertire l'algoritmo da ricorsivo a iterativo.

La prima soluzione la vedremo con un altro algoritmo di ordinamento che si chiama appunto "mer-

Figura 2.
Il grafico dell'ordine algoritmico.

ge sort", mentre la conversione a iterativo dell'algoritmo esula dai nostri scopi attuali e la rimandiamo magari ad un articolo dedicato all'argomento.

Nel grafico che abbiamo riprodotto a fianco (Figure 2 e 3), abbiamo voluto mettere in luce il fatto che l'ordine 2N è migliore, al crescere di N, rispetto anche all'ordine NLog(N), che viene considerato un buonissimo risultato per un algoritmo di calcolo.

[Sm]

Figura 3.

L'opinione

Il mondo visto dai retrocomputeristi

Sciatteria all'italiana

Sò da me che non sono una tipa che definireste "accomodante". Ho il mio carattere, "al fulmicotone" dice mio marito (ma solo perché mi vuol bene :-), ma proprio non so starmene zitta.

Dunque eravamo tutte mamme o quasi all'assemblea dei genitori alle elementari. Per chi non è addentro alle segrete cose dirò brevemente che si tratta di una aggregazione tutt'altro che spontanea dei genitori dei bambini di una certa classe che dovrebbe essere, nelle migliori intenzioni di chi l'ha voluta, cioè un certo ministro nella notte dei tempi in seguito ad una riforma da lui voluta (ogni ministro ne fa una, lo sappiamo), un momento di condivisione di problemi, ricerca di soluzioni, conoscenza reciproca. Il tutto ben inteso per il bene di coloro che sono "la luce dei nostri occhi", cioè, sempre per coloro che non "praticano", dei nostri pargoli.

Di cosa si parla in codeste congreghe di mamme? Qualche raro papà a volte viene, poi smette di venire quando si accorge che l'idea della scuola al femminile è di molto lontana dalla sua percezione e dai suoi interessi, che sono due: il calcio e la fica, citati non in ordine visto che per il maschio la priorità la merita la cosa che non è stata goduta per ultima, sia essa la partita di champion che quell'altra cosina. Del resto di che altro dovrebbe interessarsi proficuamente una mente geneticamente limitata?

Visto che degli argomenti tabù non si può parlare, altrimenti la maestra ti sbatte fuori dalla scuola assieme a tuo figlio, non rimane che parlare dei tre pilastri della scuola italiana moderna: la mensa, l'inglese e l'informatica, detto proprio nell'ordine di importanza.

Per inciso e per soddisfare la vostra legittima curiosità gli argomenti tabù sono proprio quelli dei quali varrebbe la pena parlare, sempre nell'interesse della Scuola, con la S maiuscola. Vediamoli assieme, se vi regge lo stomaco: l'insegnante sciattona che non ha voglia di insegnare nulla perché "...questo è l'ultimo anno, poi vado in pensione". C'è poi quel certo zingarello (ma ora mi si dice che è politically correct chiamarli Rom), che è venuto in classe con un cellulare dichiarando tranquillamente che l'aveva rubato giusto prima della campanella ad una anziana e con un coltello a serramanico con lama di dieci centimetri che gli ha regalato suo zio, forse per liberarsene in occasione di una perquisizione,

dico io.

Per carità: ne va della nostra anima, tutta tesa a favorire l'integrazione. Da parte mia ho deciso che quando il caro zaingarello si presenterà con una pistola, io manderò mio figlio a scuola con il giubbotto antiproiettile, poi vediamo.

Ce ne sarebbe abbastanza per giustificare il titolo di questo pezzo, ma non è di questo argomento che volevo parlare, ma della famosa "Informatica" e di come essa sia concepita alle elementari.

Dunque mio figlio (10 anni) si arrangia da solo con il computer: capisco che è conseguenza del fatto che se li ritrova in casa da quando è nato. Tranquillamente installa programmi, applica aggiornamenti che si va a cercare su Internet; consulta la Wikipedia se vuol conoscere il significato di qualche cosa e si ritocca da solo le foto fatte con la sua macchina digitale.

Non è un caso isolato, altri due o tre bambini della sua stessa classe non hanno bisogno di fare esercizi elementari come cliccare con il mouse sul triangolo verde o sul quadrato rosso. Ma si capisce che la scuola deve essere una opportunità per tutti e quindi è giusto che coloro che non lo hanno mai visto ne possano imparare le basi con una guida esperta.

Ma perché ostinarsi a considerare tutti i bambini allo stesso livello e non valorizzare invece le capacità dei pochi che conoscono bene lo strumento e che potrebbero essere loro una guida, molto più confidente di una maestra, per i loro compagni?

Mio figlio ha preso una nota sul "libretto personale" perché in sala computer aveva lanciato Word, scritta la frase banale proposta per esercizio e poi stampata nel tempo in cui la maestra si accingeva a spiegare dove si trova l'icona di Word e come si esegue il programma. "Disturbava in classe" diceva la nota e la spiegazione verbale della maestra è stata candidamente che lei non aveva dato ancora l'ordine di partire e che non si doveva stampare perché se la stampante si inceppa poi il tecnico impiega un mese a ripararla. Volevo chiederle chi mai era questo genio di tecnico informatico, potrei averne bisogno :-). Non mi preoccupo certo, sono dei malintesi, ve ne saranno altri e la povera insegnante ha diritto anche lei di guadagnarsi il pane.

Quello che mi preoccupa sono i genitori: qualcuno, anzi buona parte, dichiara che "lui non va d'accordo con il computer", come se questi fosse un essere umano dotato di raziocinio e di idee proprie!

Mamme, svegliatevi! Genitori attivatevi! Come è possibile che nel terzo millenio ci sia gente che passa il proprio tempo davanti alla TV piuttosto che cercare di imparare qualcosa di nuovo?

Non vi piace il computer? Sta bene, ma con l'inglese come stiamo?

Sciatteria all'italiana.

[Alb]

Apple Club

Tutti i linguaggi dell'Apple (7)

La mela come paradigma della programmazione

```
*************************************
**   FORTH ][ VERSION 3.2          **
**                                 **
**   SEPTEMBER, 1979               **
**   ALL RIGHTS RESERVED           **
**                                 **
**   WILLIAM G. GRAVES             **
*************************************

     FORTH FLOATING POINT ROUTINES
            ALFRED CLARK, JR.
        UNIVERSITY OF ROCHESTER
              AUGUST 1980

TO ACTIVATE THE FLOATING POINT
INTERPRETER, TYPE CONTROL-F, RETURN.

#
#
```

Ormai quasi nessuno usa più il linguaggio FORTH e sono pochi coloro che possono dire di averne avuto un assaggio. La programmazione visuale, necessaria con l'avvento delle interfacce grafiche, ha concentrato lo sviluppo dei linguaggi nella sfera della programmazione ad oggetti, estensione naturale di quel paradigma di programmazione ad eventi, così necessario quando si ha a che fare con i click del mouse!

Così ora si lavora in Java o al massimo in C++, con tutte le loro varianti; oppure si continua, i pochi appassionati, ad usare Visual Basic e il sempre stimato Delphi della Borland, anche se ormai sono guardati con condiscendenza, come si avesse a che fare con vecchi vetusti e forse un po' idioti.

All'epoca gloriosa (mi si passi il termine) dell'home computing, quando anche un piccolo incremento di velocità ed di efficienza era perseguito come un dovere morale, la ricerca di metodi nuovi per programmare quella macchina infernale che cominciava solo allora a mostrare i primi barlumi delle sue capacità, la ricerca di nuovi linguaggi era un campo molto fiorente.

Il fatto è che la programmazione non era uscita ancora dall'aurea dei centri di calcolo e di ricerca (scarsi) che esistevano nel mondo. Dicendo "nel mondo" in realtà si deve intendere "negli Stati Uniti",

Il FORTH è uno dei tantissimi linguaggi di programmazione disponibili per la piattaforma Apple II.

visto che altro posto non esisteva, almeno allo stesso livello. Così, accanto alla tendenza a scoprire "il linguaggio" per eccellenza, si affiancava un fiorente campo di adattamento di quanto disponibile su sistemi più grandi e costosi. In qualche modo, e a ragione, si pensava che la disponibilità di un particolare linguaggio, avrebbe arricchito il sistema e avrebbe catturato una sfera sempre più ampia di utilizzatori professionali.

Peraltro il BASIC mostrava tutta la sua limitatezza, al punto che nessuno credeva fosse sopravvissuto all'epoca pionieristica, dato che le nuove frontiere della semantica dei linguaggi si stava facendo strada.

Il FORTH fa parte di questa categoria di linguaggi "speranzosi" e pertanto trasportato in molti dialetti anche sull'Apple II.

Il linguaggio non è né bello nè facile da imparare, ma ha una intima connivenza con la struttura dei processori e ne ricava da ciò una marcia in più in termini di efficienza esecutiva, cosa che l'ha reso popolare per un certo periodo.

Citiamo dalla Wikipedia:

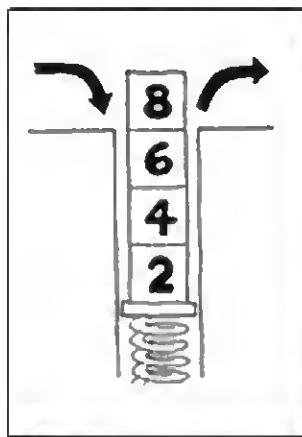
"Il Forth è un linguaggio di programmazione sviluppato da Chuck Moore presso l'US National Radio Astronomy Observatory (NRAO) durante gli anni '60 (ma che venne formalizzato solo nel 1977, per essere poi standardizato dall'ANSI nel 1994). Il nome deriva dal fatto che Moore pensava fosse un linguaggio di quarta generazione (fourth generation language in inglese), ma il computer sul quale il sistema venne sviluppato ammetteva solo nomi di cinque lettere."

Come primo esempio di implementazione sull'Apple, abbiamo scelto quella sviluppata da un certo "William G. Graves", chiamata FORTH][, nella versione 3.2 rilasciata nel settembre del 1979 nell'ambiente DOS 3.3.

Questa implementazione gode di una certa popolarità per essere stata la prima ad inserire due importanti estensioni: l'aritmetica floating-point e l'alta risoluzione grafica.

Il linguaggio

Prima di tutto alcune nozioni elementari del linguaggio. Il Forth si fonda sull'idea dello stack: basta immaginare di avere disponibile questa struttura dati organizzata LIFO (Last In First Out) e di operare su essa con i consueti operatori. Il Forth è molto "forte" nei calcoli e lo fa con estrema efficienza.

La sua naturale predisposizione è l'interprete interattivo: si inseriscono dati e operatori e si ottengono immediatamente i risultati, il tutto avvalendosi della logica a stack LIFO, la stessa che viene usata ad esempio nelle calcolatrici HP, chiamata anche Notazione Polacca Inversa o "postfix notation".

Ecco come funziona lo stack, come un distributore di piatti in un ristorante: si prende sempre l'ultimo impilato.

Vediamo un esempio:

per sommare due numeri, diciamo 3 e 4 si procederebbe, nella nostra normale notazione aritmetica: 3 + 4 =

Invece nel Forth i tasti da digitare in sequenza sono:

3 4 +

Alcuni caratteri speciali e comandi permettono di manipolare lo stack. Il più semplice operatore è "." (un semplice punto) che estrae il numero in cima allo stack e lo stampa a video.

Ad esempio:

1 2 3 4 . . . .

stampa 4 3 2 1

Un altro operatore particolarmente comodo è DUP, che duplica la cima dello stack. Per ottenere il quadrato di un numero, diciamo 3, si opererebbe con la sequenza:

*3 DUP * .*

Purtroppo non si può dire che il Forth favorisca la chiarezza del codice, ma bisogna dire che una trentina di anni fa questa cosa appariva meno urgente, forse perché il codice in giro era pochino e normalmente abbastanza elementare.

Non ci sembra il caso di approfondire troppo il linguaggio, cosa che ci impiegherebbe ben più del modesto spazio cui disponiamo. Lo proporremo forse come argomento di una serie di articoli sui retro-linguaggi.

Altre implementazioni

Questa versione del Forth sotto DOS 3.3 sull'Apple II non è naturalmente l'unica. Quando mai esistono cose uniche? Ne troviamo una più evoluta che gira sotto PRODOS ed è compliant con le specifiche dell'interprete FIG Forth (FIG è un acronimo di Forth Interested Group, una organizzazione che da sempre ha portato avanti iniziative per la standardizzazione e diffusione del linguaggio).

Le migliorie che ci aspettiamo dall'adozione del nuovo sistema operativo di Apple ci sono tutte, compresa, come si vede dalla figura, l'uso delle ottanta colonne per

Stiamo lavorando con l'interprete. Il simbolo "#" è il prompt.

```
##
#
#
#
#
#
#
#
#
#
#
#
#
#
#3
#4
#+
#.
7#
#3
#DUP
#*
#.
9#
```

il video.

Esiste anche una versione sotto CP/M, si tratta del CPM-FORTH contenuto in un unico supporto, purtroppo senza molta documentazione. Quello che ne abbiamo ricavato è che in realtà si tratta di un compilatore in grado di interpretare e convertire in codice oggetto dei sorgenti che "assomigliano" a codice Forth. Questo è tutto quello che siamo riusciti a ricavare dalle nostre, per la verità non troppo approfondite indagini.

Pro-Forth sotto PRODOS.

[Sm]

```
A>dir
A: DEBUG    COM : DEV     VOM : SL5    COM : ASSEM   COM
A: SYSMAKE  CDM : CDLD    SL5 : SL5    DOC : CDMMDD  SL5
A: DEBUG    SL5 : KERNEL  SL5 : ASSEM  SL5 : ICDND   SL5
A: MUSIMP   $$$
A>type kernel.sl5


          ( Copyright 1980 The Stackworks.  All rights reserved )

                    ( Z80 VERSIDN 1.2  -  CP/M )
                            ( 3/25/80 )



CDDE $$PUSH    HL PUSH   EXX  (HL) C LD  HL INC  (HL) B LD
               HL INC  BC PUSH  EXX  RET    EDOC


CODE $CONSTANT    HL POP   (HL) E LD   HL INC   (HL) D LD
                  DE PUSH   $NEXT JP    EDOC

CODE $BARRAY   HL POP   (HL) E LD   HL INC   (HL) D LD   HL POP
               DE HL ADD   $PUSH JP     EDOC

CDDE $ARRAY    HL PDP   (HL) E LD   HL INC   (HL) D LD   HL POP
               HL HL ADD   DE HL ADD   $PUSH JP     EDOC


( ====================================================================== )

              ( CONDITIONAL CDMPILATION CONSTANTS )

        1 XCONSTANT ?FILESYS    ( COMPILE FILE SYSTEM SECTION         )
        0 XCDNSTANT ?DWNIO      ( CDMPILE USER I/D SECTION            )
        1 XCONSTANT ?CRT        ( COMPILE CRT I/O WORDS               )
        1 XCONSTANT ?INTERP     ( COMPILE INTERPRETER SECTIDN         )
        1 XCDNSTANT ?CDMPILE    ( CDMPILE THE CDMPILER SECTIDN        )

( NOTES:  [1]  CRT SECTION NEEDS OWNIO OR FILESYS                      )
(         [2]  INTERPRETER NEEDS CRT SECTION                           )
(         [3]  CDMPILER NEEDS INTERPRETER TO RUN.                      )
(         [4]  DO NOT COMPILE BOTH OWNIO AND FILE SYSTEM.              )

( ====================================================================== )
```

Qui abbiamo riportato il DUMP di una sessione di lavoro sotto CP/M. L'azienda che commercializza questa versione del linguaggio si chiama con poca fantasia "StackWorks". In effetti il FORTH è tutto "stack"!

BBS

Pubblicare una fanzine

A colloquio con i lettori

Vorrei tornare sul tema della pubblicazione di una rivista/fanzine, dal punto di vista organizzativo e di impegno.

L'editoriale del mese scorso, nel quale se ricordate accennavo alla cosa, è stato oggetto di qualche richiesta di chiarimento e di intervento da parte dei nostri lettori. Sinceramente ne sono rimasto sorpreso: non pareva proprio che ci fosse questo grande interesse a pubblicare, invece non è così. Evidentemente molta più gente di quanto si pensi ha qualcosa da dire e vorrebbe farlo in una forma il più possibile "classica", come appunto un libro, una rivista, un bollettino,...

Pensandoci bene non è poi così sorprendente che ci sia tanto interesse alla pubblicazione in genere: ci sono centinaia di scrittori "in erba" solo in Italia, forse anche qualche migliaio, che premono alla porta degli editori sperando di sfondare. Così anche per la professione del giornalista, ora immensamente e secondo me anche immeritatamente osannata dai media in tutte le forme. Cavolo, basta leggere delle statistiche sulla frequenza di facoltà universitarie quali "Scienze della comunicazione" per capire che buona parte degli italiani, in maggioranza donne, vorrebbero intrapprendere quella strada.

Stendiamo un pietoso velo su gran parte di detta popolazione e sul metodo che essa adotta per arrivare "in cima"; ci porterebbe polemicamente lontano...

Dunque siamo un popolo di scrittori. Ma non bastano i blog per soddisfare il "prurito di penna"? Per molti certo questa è la strada più facile, anche se immensamente inflazionata. Io credo che esistano blog che non sono mai stati visitati da nessuno se non lo stesso proprietario! Se questi lo aggiorna per un suo diletto personale, va benissimo, ma so che molti ci rimangono male e non si spiegano la mancata frequentazione delle masse. Questi sono i polli che cadono in bocca ai "lupi" del "ti vendo il posizionamento nei motori di ricerca", come se poi fosse questa la cosa più importante invece che il contenuto.

Molti scrittori pagano di tasca propria gli editori per venire pubblicati. Non solo, si accollano l'acquisto di metà tiratura che poi non sanno come liberarsene. Gli amici di questi ricevono la stessa copia del libro per tre Natali di seguito...

Per dare un'idea so che presso

un editore abbastanza noto, anche se non al vertice della categoria, la stampa con coperta cartonata di un 200 pagine, costa allo scrittore diecimila Euro. Gli editori si guardano bene di "tirare" molte copie, tanto possono sempre ristampare. Così mettono in produzione 200 copie del libro, 100 delle quali le "deve" comprare l'autore, diciamo altri 10.000 Euro perlomeno. Così l'editore, ce ne sono alcuni che fanno bilancio in questo modo, incassa 20.000 Euro impegnandosi pochissimo. Un affare!

E veniamo alle riviste, chiamiamole così, anche se "fanzine" sarebbe una definizione migliore per inquadrare la tipologia di pubblicazioni di cui trattiamo. Ce ne sono moltissime in Italia e un numero sterminato nel mondo e riguardano tutti i possibili e immaginabili argomenti. Il fumetto la fa da padrone, così come le fanzine di poesia e letteratura: lo si capisce bene perché sono le categorie di autori più frustrate in assoluto dalla difficoltà di essere letti e pubblicati. Subito a seguire le cosidette "underground fanzine", quelle che trattano di un mondo diciamo "alternativo", di protesta, magari identificandosi con un genere musicale "duro".

Di fanzine dedicate al retro computer ce ne sono poche: o si tratta di iniziative portate avanti da musei e che quindi ne costituiscono un canale di comunicazione per le iniziative che l'organizzazione intraprende, o sono monotematiche, dedicate cioè ad un singolo computer e/o marca. Lo si capisce benissimo per il fatto che se si devono aggregare degli interessi comuni, è più facile farlo per persone che condividano lo stesso interesse per un a certa piattaforma di calcolo. A parte metterei le iniziative che si occupano di retro-gaming. Non perché esso sia un tema disdicevole, ma semplicemente perché mi sembra vadano a costituire una goccia nel mare: è un argomento molto vasto e fare delle recensioni di giochi mi sembra un lavoro improbo, oltre che esasperatamente settoriale, seppure con una sua importanza storica.

Dunque, derimati gli stimoli che spingono le persone a tentare la strada, come ci si organizza?

La prima cosa da dire è che non occorre essere in molti, anzi si potrebbe fare tutto da soli, avendone le capacità e disponendo del tempo necessario. In questo caso la scelta di un particolare settore di interesse appare inevitabile dal momento che bisogna pur intendersene un pochino di quello che si va a comunicare, non siamo mica giornalisti :-).

L'equazione "più siamo meglio è" non è affatto esatta. Per riuscire nell'impresa la regola fondamentale è quella di copiare l'organizzazione delle iniziative commerciali. Prima di tutto ci vuole un leader, cioè una persona in grado di mettersi alla testa e trascianare gli altri. Questi deve decidere, seppure sentendo il parere di tutti, comunque una decisione deve essere

presa e non può essere "a maggioranza". Le aziende non funzionano in maniera democratica!

In questo senso l'ideale sarebbe avere un'unico centro di decisione e una lista di collaboratori dai quali pescare gli articoli. Questa via viene seguita ad esempio da due riviste attive nel mondo Linux: Linux&c, con il deux ex machina Tassone e Linux Pro con il simpatico Massimo Zagaglia. Questi scelgono di volta in volta cosa pubblicare, potendo pagare i collaboratori e quindi permettendosi di guidarne la scrittura o semplicemente respingerne gli sforzi e le idee.

Ovviamente con una iniziativa amatoriale "fra amici" questo non è possibile; è necessario che ognuno possa identificarsi con il periodico, lo senta suo, almeno per le parti che cura personalmente.

Organizzativamente non posso certo dirvi cosa è meglio, posso semplicemente raccontarvi come siamo organizzati noi e perché JN funziona.

Noi siamo in quattro: Tullio, Sonicher, Lorenzo e Salvatore; Bes è della partita ma non è proprio un appassionato di retro computing. Ognuno segue uno o due settori in particolare, anche se abbiamo ampia libertà di muoverci per realizzare, se non proporre, articoli che sconfinano dal normale campo di interesse di ognuno.

Uno di noi, a turno, tiene le fila dell'organizzazione. L'ha fatto Tullio per due anni, ora tocca a me, Salvatore. Il cosidetto "leader" decide le date di pubblicazione e la composizione della rivista, scrive l'editoriale, risponde alle e-mail che arrivano e tiene i vari contatti.

Il fascicolo viene deciso molto in anticipo in quasi tutti gli argomenti, almeno due mesi prima della data di rilascio, anche se a volte rimangono dei "buchi" fino all'ultimo minuto.

Ogni collaboratore fisso possiede i template delle pagine che ospitano il suo contributo e il suo compito è consegnare tale pre-elaborato già completamente impaginato, con font esatti, colori a posto, immagini decise nella dimensione e disposizione. Come software usiamo Indesign della Adobe. Questi ha il vantaggio di essere compatibile su due piattaforme: Mac OS X e Windows.

In questo modo si riduce enormemente il lavoro di impaginazione e si evita che una persona debba passare tutto il tempo a mettere in ordine i vari contributi. Nonostante questo l'ottenere il fascicolo completo richiede comunque circa una settimana di lavoro: sono molte le cose che rischiano di andare storte, per non parlare della fase di correzione: per quanto si legga e rilegga, qualche refuso sfugge sempre.

Qualcuno mi ha chiesto perché consiglio di partire solo quando si hanno a disposizione contributi per almeno tre fascicoli completi. La risposta si basa sul concetto di "invaso completo". Quando si

progetta una centrale idroelettrica alimentata da un bacino acquifero, prima di partire con la produzione si ha cura che l'invaso sia pieno, sfruttandone così al massimo le capacità di "volano" o "buffer".

Così è anche per una rivista: mentre escono i fascicoli già preparati, arrivano altri contributi che troveranno posto nei fascicoli a venire e non necessariamente nell'ordine di arrivo.

Un'altra esperienza che abbiamo fatta è che la periodicità troppo stringente è impossibile da rispettare. Per fare un mensile di cinquanta pagine occorrerebbero almeno dieci contributors, oltre che un impegno inconciliabile con la necessità di lavorare per mantenerci. Come vedete, partendo dai sei numeri/anno, abbiamo anche noi cercato di crescerne la frequenza, ma dobbiamo rinunciare e anzi vi comunico che torneremo con gennaio 2009 alla periodicità più o meno bimestrale.

Quanto deve essere "grosso" il fascicolo? Altro argomento "spinoso" perché una iniziativa di poche pagine rischia di passare inosservata, mentre una ricca di argomenti di essere eccessivamente costosa da realizzare. Gli editori che pubblicano in cartaceo hanno un criterio basato su costi/benefici calcolati con dei parametri che tengono conto dei costi di stampa, di distribuzione, etc... Quasi tutto è però legato alla pubblicità che si vuole raccogliere. Noi non abbiamo questi tedii: non stampiamo e siamo distributori di noi stessi, perciò ci possiamo sbizzarrire in proposito. La dimensione dipende allora da quanti articoli si riescono a produrre fra una uscita e l'altra. Non bisogna comunque "buttarsi" a pesce al momento della partenza per poi perder progressivamente l'entusiasmo: meglio impegnarsi ognuno a produrre una quantità minimale di materiale, ma in maniera costante.

Il lavoro di evoluzione della grafica e layout è paradossalmente quello che ci preoccupa di più. Per i contenuti ci pensiamo noi, ma costruirsi una solida competenza di grafica ed editoria è un vero scoglio. Per fortuna ci da una mano Anna, una amica che lo fa di mestiere, anche se non possiamo chiedere che lavori a tempo pieno per noi. Finora l'impaginazione originale con qualche correzione di rotta sono stati sufficienti a tenere botta, per il futuro cercheremo di perseguire quello che è stato fatto finora: una lenta ma costante marcia di affinamento con pochi ma mirati interventi su layout e grafica (avete visto il nuovo logo multicolore?).

Termino qui questa disgressione sull'arte della pubblicazione sperando di aver contribuito a chiarire le idee dei nostri amici che hanno in testa di dare vita ad una fanzine e svelandovi anche qualcosa in più di noi e dei nostri piccoli segreti.

[Sm]

L'Apple //c sarà il protagonista al centro della scena del fascicolo numero 21.